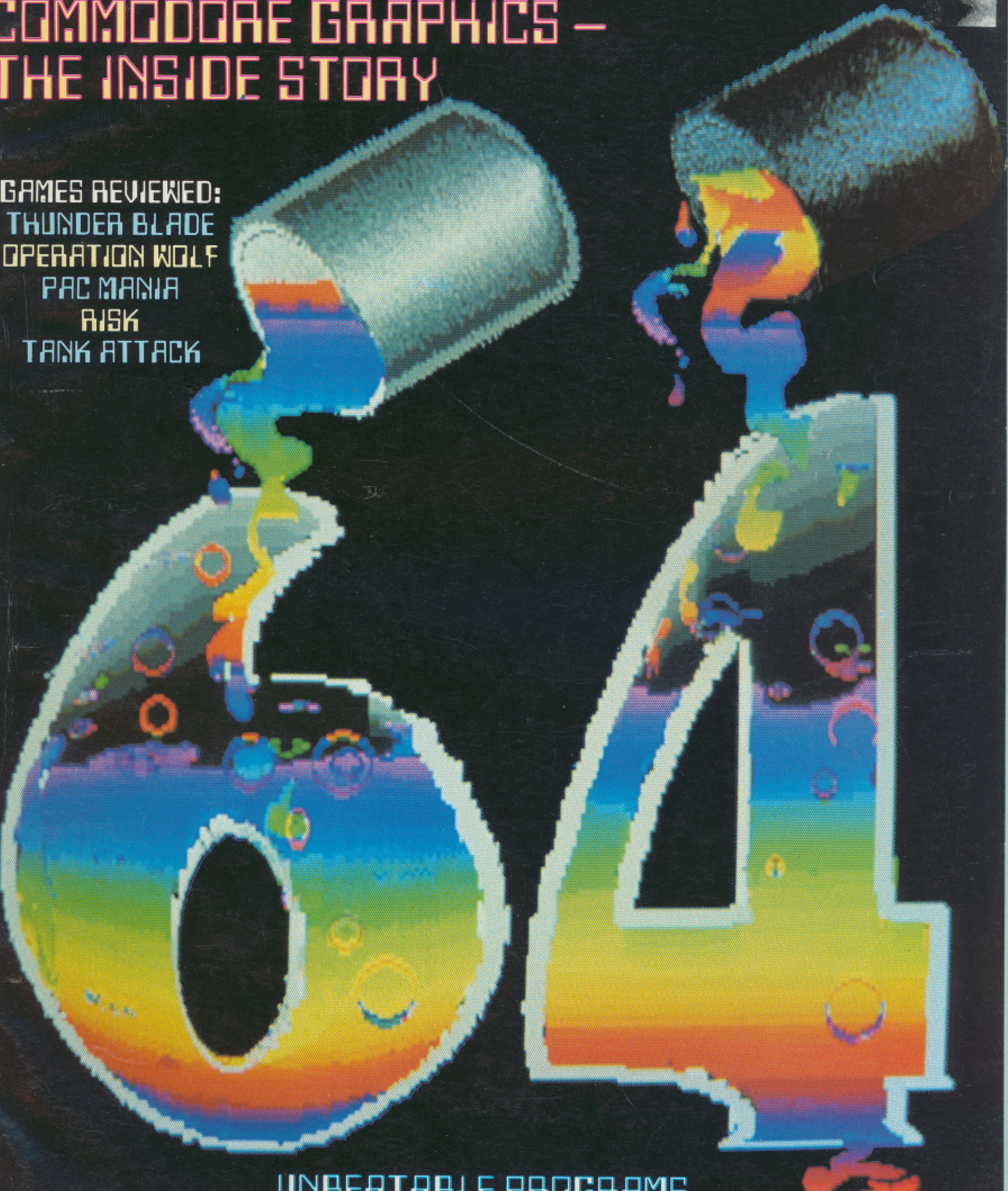
YOUR

COMMODORE

MARCH 1989 £1.30

COMMODORE GRAPHICS – THE INSIDE STORY

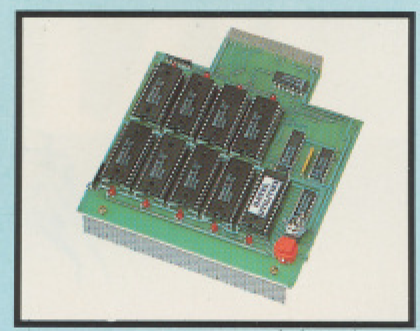
GAMES REVIEWED:
THUNDER BLADE
OPERATION WOLF
PAC MANIA
RISK
TANK ATTACK



UNBEATABLE PROGRAMS

LETTER WRITER ▲ BASIC WORKSHOP ▲ PERSONAL FILE

DATEL ELECTRONICS



256K SUPEROM EXPANDER

- Select instantly from 8 sockets which accept up to 32K EPROM each.
- Program your own EPROMs using our EPROM programer.
- No need to have lots of cartridges - just make a selection from the Superom menu.
- Fully menu driven on power up.
- Select any slot under software controls.
- Unique EPROM generator feature will take your own programs - basic or m/c & turn them into autostart EPROMs. (EPROM burner required).
- Accepts 2764/27128/27256 EPROMs.
- On board unstoppable reset.
- On board operating systems - no programs to load.

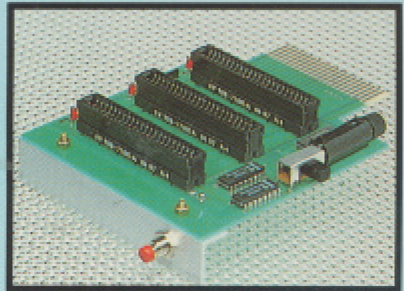
ONLY £29.99



RESET CARTRIDGE

- Unstoppable reset button. NOTE:- user port cheaper type reset buttons offered by others are not unstoppable.
- Resets even so called "unstoppable" programs.
- Add pokes from magazines etc.
- Fits in cartridge port.
- Simply plug in.

ONLY £5.99

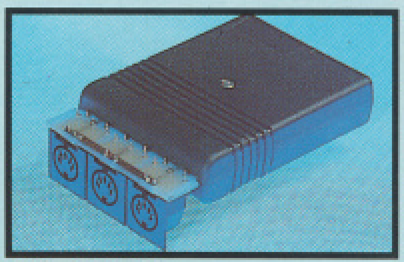


3 SLOT MOTHERBOARD

SAVE WEAR & TEAR ON YOUR EXPANSION PORT

- Accepts 3 cartridges.
- Switch in/out any slot.
- Onboard safety fuse.
- Fully buffered.
- High grade PCB.
- Reset button.

ONLY £16.99



MIDI 64

FULL MIDI INTERFACE FOR THE 64/128 AT A REALISTIC PRICE.

- MIDI in/MIDI thru/2 x MIDI out.
- Compatible with most leading software packages.

ONLY £29.99



DIGITAL SOUND SAMPLER

- The new sampler allows you to record any sound digitally into memory & then replay it with astounding effects.
- Playback forwards/backwards with echo/reverb/ring modulation.
- Now with full sound editing module to produce outstanding effects.
- Full 8 bit D to A & ADC conversion.
- MIDI compatible with suitable interface. (i.e. Datel unit for £29.99, see ad.).
- Live effects menu includes real time display of waveforms.
- Line in/mic in/line out/feedback controls.
- Powerfull sequencer with editing features.
- Load/save sample.
- Up to 8 samples in memory at one time.
- Tape or disk (please state).
- Complete software/hardware package.

ONLY £49.99

COM-DRUM

DIGITAL DRUM SYSTEM

- Now you can turn your digital sound sampler into a digital drum system.
- 8 digital drum sounds in memory at one time.
- Complete with 3 drum kits.
- Real drum sounds - not synthesised.
- Create superb drum rhythms with real & step time.
- Full editing. Menu driven.
- Output to hi-fi or through TV speaker.
- Load/save facilities.

ONLY £9.99
STATE TAPE OR DISK

COM-DRUM EDITOR

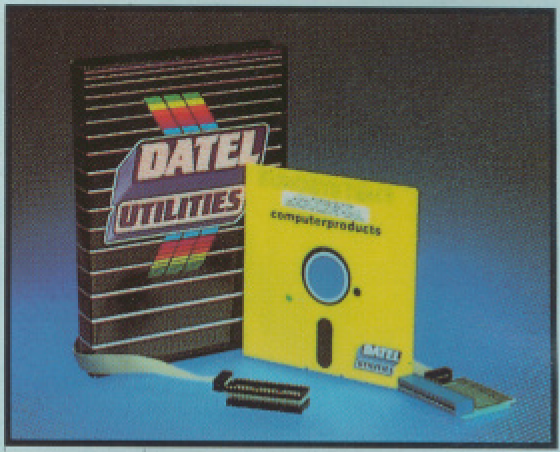
- 24 drum sounds supplied on disk to enable you to construct your own drum kit.
- Re-arrange sounds supplied with a Com-Drum to make new kits.
- Load/save facilities.

ONLY £4.99 DISK ONLY

DATA RECORDER

- Quality Commodore compatible data recorder.
- Pause control, counter, etc.
- Suitable for 64/128
- Send now for quick delivery.

ONLY £24.99



DEEP SCAN BURST NIBBLER™

- The most powerfull disk nibbler available anywhere, at any price! Burst Nibbler is actually a two part system - a software package & a parallel cable to connect the 1541/1570/1571 to 64/128 (state type).
- What gives Burst Nibbler it's power? Conventional nibblers have to decode the data from the disk before it can transfer it using the serial bus - when non standard data is encountered they are beat. Burst Nibbler transfers data as raw GCR code via the parallel cable without the need to decode it so you get a perfect copy of the original.
- ● Will nibble up to 41 tracks ● Copy a whole disk in under 2 minutes ● Full instructions ● Regular updates - we always ship the latest ● Fitted in minutes - no soldering usually required ● Full or 1/2 tracks ● No need to buy parallel cable if you have Disk Demon/Dolphin etc. ● Cable has throughbus extension for other add ons.
- Wether to choose FastHack'em or Burst Nibbler? Fast Hack'em is unbeatable value as an "all rounder" - with nibblers, 1 or 2 drive copy, format, file copy, 1571 copy etc. etc., so if you have a more general requirement perhaps Fast Hack'em is for you. Burst Nibbler is a pure nibbler second to none, for the reasons stated. So if it's just making backups you are interested in, there is no other product to beat it!

ONLY £24.99 COMPLETE

SOFTWARE ONLY £12.99 **CABLE ONLY £14.99**



ROBOTEK 64

MODEL & ROBOT CONTROL MADE EASY.

- 4 output channels - each with onboard relay.
- 4 input channels - each fully buffered TTL level sensing.
- Analogue input with full 8 bit conversion.
- Voice input for voice control.
- Software features:- test mode/analogue measurement/voice activate/digital readout etc.

ONLY £39.99

INCLUDING HARDWARE/SOFTWARE/MIC. ETC,(STATE TAPE OR DISK).

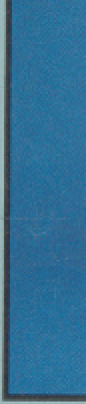
TOOLKIT IV

THE ULTIMATE DISK TOOLKIT FOR THE 1540/1541.

- A disk toolkit is an absolute must for the serious disk hacker. Toolkit IV has more features than most for less.
- DISC DOCTOR V2 - Read & write any track & sector including extra & renumbered tracks. Repair damaged sectors. Look underneath read errors.
- HEADER/GAP EDITOR - Decodes & displays ALL header information including off bytes & header gap. Rewrite the entire header & header gap. Renumber sectors. Also edit any sector tail gap.
- DISK LOOK - Sort directory. Recover lost files. Display file start /end addresses. Disassemble any file program directly from the disk to SCREEN or PRINTER including undocumented opcodes. Edit Bam.
- FILE COMPACTOR - Can compact machine programs by up to 50%. Save disk space. Compacted programs run as normal.
- FAST DISK COPY - Copy an entire disk in 2 minutes or less using single 1541.
- FAST FILE COPY - Selective file copy. Works at up to 6 times normal speed.
- FORMATTER - 10 second format an entire disk or format any individual track or half track 0 to 41. Re-define any of 30 parameters to create or recreate unique disk formats.
- ERROR EDIT - Quickly find & recreate all read errors including extras & renumbered tracks or sectors & half tracks from 0 to 41. Even recreates data under errors & allows you to redefine any necessary parameters.

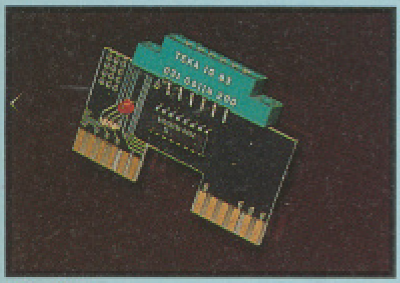
ONLY £9.99







DATEL ELECTRONICS



NEW CLONEMASTER

- The most effective tape to tape back-up board available.
- Makes perfect backups of your tapes easily & effectively!!
- No user knowledge needed at all.
- On board TTL logic circuitry actually shapes the program & sends a perfect signal to the record cassette - producing a copy better than the origional in many cases.
- L.E.D. indicator shows when data is being transferred to avoid exessive tape winding.
- Works with almost any program including multi-loaders, turbos & even very unusual type turbos.
- Requires access to two CBM compatible data recorders.
- Simply press 'Play" on one recorder & press 'Record' on the other - that's it!
- You can even make a backup while you are loading the program.
- This is a total hardware solution - no programs to load - the results are stunning!

ONLY £9.99



TURBO ROM II

TURBO ROM II IS A REPLACEMENT FOR THE ACTUAL KERNAL INSIDE YOUR 64. IT PROVIDES SUPERFAST LOAD/SAVE ROUTINES.

- Loads most programs at 5-6 times normal speed.
- Saves at 5-6 times normal speed as well.
- Improved DOS support including 10 sec. format.
- Programed function keys:- load, directory, old, etc.
- Return to normal kernal at a flick of a switch.
- FCOPY-250 block file copier.
- FLOAD-special I/O loader.
- Plus lots more.
- Fitted in minutes - no soldering usuallly required. (On some 64's the old ROM may have to be desoldered).

ONLY £14.99



SMART CART

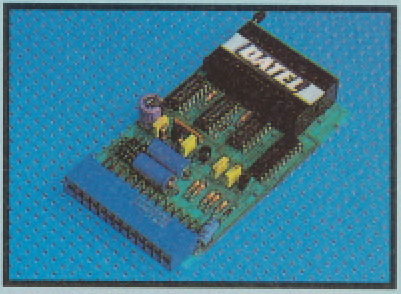
- 32K pseudo ROM.
- Lithium battery lasts up to 5 years.
- Simply load the program you require - then flick the switch. The cartridge can then be removed just like a ROM cartridge.
- Make your own cartridges including autostart types - without EPROM burner. 32K version = 4 x 8K pages.
- Some knowledge of M/C is helpful - but full instructions included.
- I/O 2 slot open for special programing techniques.

ONLY £29.99

RAM DISK

- Turn your Smart Cart into a 32K Ram/disk.
- 32K of instant storage area for files/programs.
- Disk type commands:- load, save, directory, scratch.
- Program data retained when computer is switched off!
- Full command set with instructions.

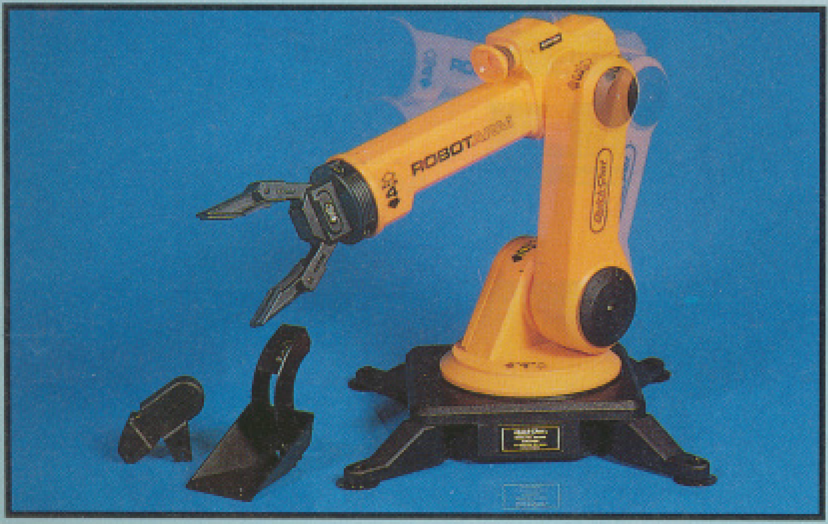
ONLY £9.99



EPROMMER 64™

- A top quality, easy-to-use EPROM programer for the 64/128.
- Fully menu driven software/ hardware package makes programing/reading/verifying/ copying EPROMs simplicity itself.
- Will program 2716, 2764, 27128 & 27256 chips, 12.5, 21 or 25 volts.
- Fits into user port for maximum compatibility with cartridges/ Superom Board etc.
- Full feature system - all functions covered like device check/verify.
- We believe Eprommer 64 is the most comprehensive, most friendly & best value for money programer available for the 64/128.
- Ideal companion for Superom Board, Cartridge Development System, our kernal expanders or indeed any EPROM base project.
- Comes complete with instructions - plus the cartridge handbook.

ONLY £39.99 COMPLETE



ROBOTARM

- Explore the facinating science of Robotics with this full feature RobotArm.
- Human like dexterity - with 5 axis of movement it is so versatile. It can manipulate small objects with amazing ability.
- Comes with Accesories including 'Finger' Jaws, Magnetic Attachment, Shovel Scoop, 4 Stabilizing Suction Legs, etc.
- Easily controlled using 2 joysticks (any 9 pin type) or connected to your computer with our Interface/ Software to give Computer/Robotic control (see Interface Offer).
- Uses 4 HP2 batteries (not supplied) to power motor movement so uses no computer power.
- Self contained, ready to use (except batteries, joysticks).

ONLY £49.99

INTERFACE OFFER

- Unique Software/Hardware package to allow you to interface your computer with the Robotarm.
- Train mode allows you to store & then repeat movement sequences.
- Very easy to use.
- This interface isn't needed to use the Robotarm but interfacing with your computer has great possibilities.

ONLY £24.99 COMPLETE WITH CABLES

BLUE CHIP DRIVE



- External power supply for cool operation.
- Even works in 1571 double sided mode (128 mode).
- Fully C64/128 compatible.
- This drive is now probably the most compatible drive available for the Commodore. More so than even Commodores own '1541C'.
- Supplied complete with all cables - no more to buy.
- At last, a top quality drive at a sensible price.

ONLY £129.99

ADD £5.00 FOR COURIER DELIVERY

ULTRA CRUNCHER

- The ultimate cartridge based program compactor.
- Compacts by up to 30%!
- 3 compacting programs on one cartridge.
- Fast loading/saving routines.
- Full DOS support including fast format.

ONLY £12.99 on cartridge

Available as chip only for 256K Superom Board for

ONLY £7.99

FAST HACK'EM ™

- Multi Module Disk Nibbler - all on one disk.
- Single 1541 Nibbler - Copy a disk in 2 minutes.
- Super fast File Copy - under 9 seconds typically.
- Plus unique "Parameters" mode. These files "unlock" the more heavily protected programs with a parameter for a particular brand or even single program.
- No other product is such a good "All Rounder" at this price...

ONLY £9.99

ALL ORDERS NORMALLY DESPATCHED WITHIN 48 HRS

HOW TO ORDER ...

BY PHONE	BY POST	FAX
0782 744707 24hr Credit Card Line	Send cheques/POs made payable to "Datel Electronics"	0782 744292 UK ORDERS POST FREE EUROPE ADD £1 OVERSEAS ADD £3

PRICES AND SPECIFICATIONS CORRECT AT TIME OF PRESS AND SUBJECT TO CHANGE WITHOUT NOTICE

CALLERS WELCOME - Please reserve goods by telephone prior to visit.

DATEL ELECTRONICS

DATEL ELECTRONICS LTD., FENTON INDUSTRIAL ESTATE GOVAN ROAD, FENTON, STOKE-ON-TRENT, ENGLAND.

SALES ONLY	TECHNICAL ONLY
0782 744707	0782 744324

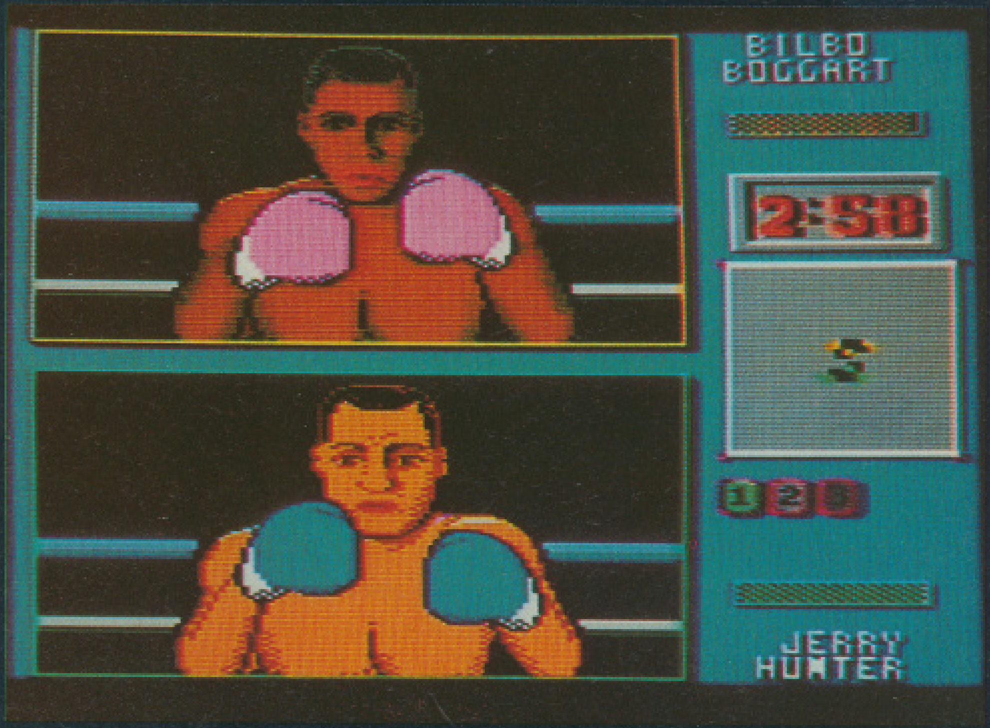
CONTENTS



Thunder Blade



Operation Wolf



TKO

VOLUME 5
NUMBER 6

FEATURES

REGULARS

PROGRAMS

Editor: *Stuart Cooke*
Deputy Editor: *Paul Whitington*
Technical Editor: *Paul Eves*
Advertisement Manager: *Paul Kavanagh*
Ad-Copy Control: *Andrew Selwood*
Artist: *Alan Batchelor*
Designer: *Neil Sweetman*
Origination: *Ebony*
Printers: *Chase Web*

Your Commodore incorporating Your 64 is a monthly magazine appearing on the first Friday of each month. Argus Specialist Publications Limited Editorial & Advertisement Office, Your Commodore, No 1 Golden Square, London W1R 3AB. Telephone: 01-437 0626 Telex: 8811896.
Subscription rates upon application to Your Commodore Subscriptions Department, Infonet Ltd, 5 River Park Estate, Berkhamsted, Herts, HP4 1HL. U.S.A. Subscription Agent: Wise Owl Worldwide Publications, 4314 West 238th Street, Torrance CA 90505 U.S.A.

ARGUS PRESS GROUP

ISSN 0269-8277

ABC

Data Statements



Radio 1 Jock, Bruno Brooks (left), does the Commodore Show

R1 DJ Meets CBM

Radio 1 disk jockey Bruno Brooks was given the works at the Commodore Christmas Show – in more than one sense. Apart from being given VIP treatment from Commodore boss Steve Franklin, Bruno was also presented with a copy of *The Works* by Brown and Wagh Direct's MD, Jim Housego.

Of special interest to the Amiga-owning DJ was the many music exhibits. Nigel Jones, Commodore's answer to Stock, Aitken and Waterman, put the MIDI-linked computer through its paces.

Who knows, maybe some of the jingles on Radio 1 may be Amiga assisted in the future.

Daisies For Epson

Printer manufacturers, Epson, have started the New year by relocating all the way to Hemel Hempstead from their Wembley offices.

The printing company's new address is Epson (UK), Ltd, Campus 100, Maylands Avenue, Hemel Hempstead, Herts HP2 7EZ. Tel: (0442) 61144.

CBM Roadshow

Commodore are going to school during the early part of this year as their education team hits the road. An initial trial was made by holding an exhibition at Middlesex Polytechnic's Bounds Green site, and it proved that there was sufficient interest from educationailists to give the green light to the roadshow.

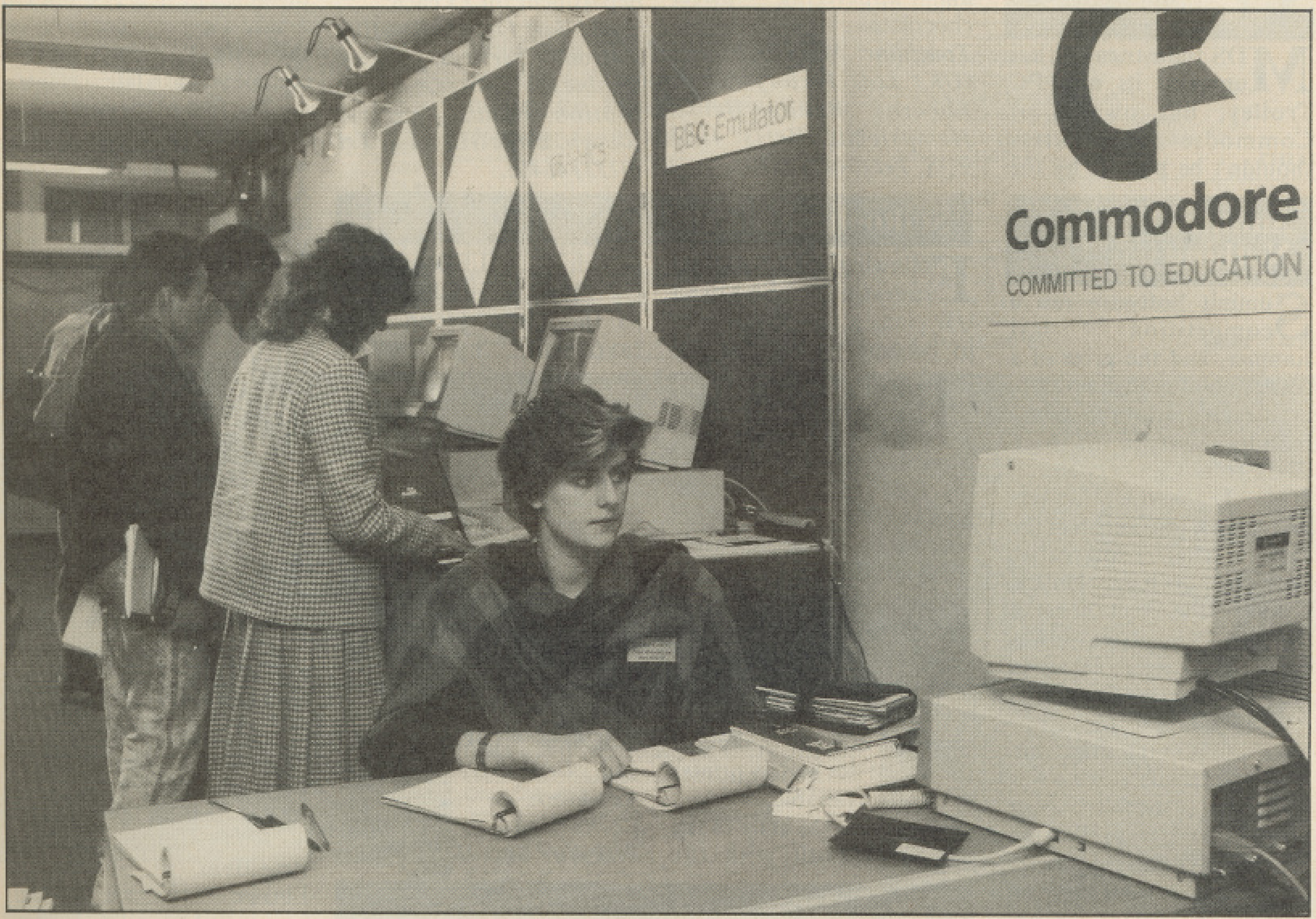
At the polytechnic, the Amiga and PC were put through their paces, and the new Acorn BBC emulator aroused a lot of interest as a possible bridge to the current machines possessed by schools.

Dates and venues for the roadshow have yet to be announced, but Peter Talbot, Commodore's National Sales Manager for Education, has high hopes – "I was delighted both with the turnout and the calibre of people attending our show. This is definitely going to become an annual event.

"Our 1988 education initiative has seen Commodore making major inroads on the education market, and 1989 will be even better."

Touchline: *CBM (UK) Ltd, Commodore House, The Switchback, Gardner Road, Maidenhead, Berks SL6 7XA.*

Commodore's Fiona Grahamslaw at the Computers in Education Exhibition



Sexist Software

OASIS is an organisation with a mission – to save the fair sex from the unfair sex in the computer world. Sandra Vogel feels that women get a raw deal when it comes to the advertising and content of computer games, and has formed the Organisation Against Sexism in Software to bring pressure to bear on the offending software houses.

This industry has no real watchdog to ensure that a standard of moral values is maintained, and this latest move has been made in response to the unfavourable portrayal of women. According to OASIS, women are mainly portrayed either as busty bimbos, or else weak creatures who must be rescued from perilous situations by macho heroes. More than this, too much software is produced to appeal to the male ego, limiting the encouragement of female participation.

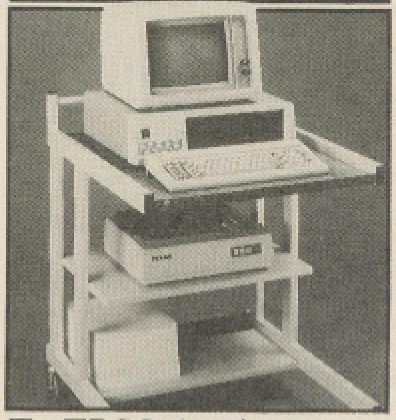
Although computer gameplaying is still very much a male preserve, the number of female computer enthusiasts is growing. Sandra does not expect special status to be given to women, merely an acceptance that they *do* exist by avoidance of the standard male/female stereotypes in advertising, and gender determination in suitable games.

OASIS already has an active membership, but is eager to recruit any Tom, Dick or Harriet who sympathises with the cause. An anual subscription costs £3, and includes six issues of the organisation's bi-monthly magazine.

One example of sexism which is heavily criticised in the first issue of the OASIS magazine is Palace Software's *Barbarian* promotion, which features the pneumatic charms of Maria Whittaker. Is it sexploitation, or merely a storm in a D-cup?

Touchline: *OASIS, Sandra Vogel, 3 Alden Court, Stanley Road, Hibledon, London SW19 8RD.*

Tower of Power



MDS Industries UK has unveiled its new PC Trolley, but owners of Commodre computers shouldn't be put off by the title, because an Amiga or its ancestors would look just as smart.

The trolley is supplied with a four-way power board, and is designed so that it can be pulled over any standard height desk or table. The range of finishes available are light grey laminte, decorative teak or an appetizing bitter chocolate.

***Touchline:** MDS Industries (UK) Ltd, Factory No 1, Ebberns Road, Hemel Hempstead, Herts HP3 9QS. **Tel** (0442) 23105.*

Summit For Nothing

Summit Software ran a competition last summer, and the prize of a collection of mint condition, pre-decimalisation coins was recently awarded to Alan Clark from Camberwell.

The competition was set to mark the launch of the Summit label by Alternative Software boss Roger Hulley. Based on the fiscal theme of the game *Sum-it*, which was devised by Roger's father in the Thirties, the competitors had to convert the £2.99 price of Summit titles into old LSD currency. Alan was the sender of the first correct entry picked from the hat and cashed in on the prize.

Two forthcoming releases from Summit are Johnson Scanatron's *The Double* (now Goliath Games) and Database's *Mini Office*. Summit's full-price sister label, Again Again, has released a game based on last year's trouble in the Arabian Gulf. *Operation Hormuz*, programmed by Durrell, costs £9.99 on the C64, but the Amiga version won't follow until later this year.

***Touchline:** Summit Software, Units 3-6, Baileygate Industrial Estate, Pontefract, West Yorks WF8 2LN. **Tel:** (0977) 797777.*

Artful Clues

Electronic Arts has published a series of detailed clue books for its ever-growing range of roleplaying adventures. *Deathlord, Wasteland, The Mars Saga*, and *Bards Tale I, II*, and *III* players can buy the books from EA's Customer Services department for £5 each.

***Touchline:** Electronic A Rts, Customer Services, Langley Business Centre, 11-49 Station Road, Langley, Slough, Berks SL3 8YN. **Tel** (0753) 46465.*

Seikosha Assault

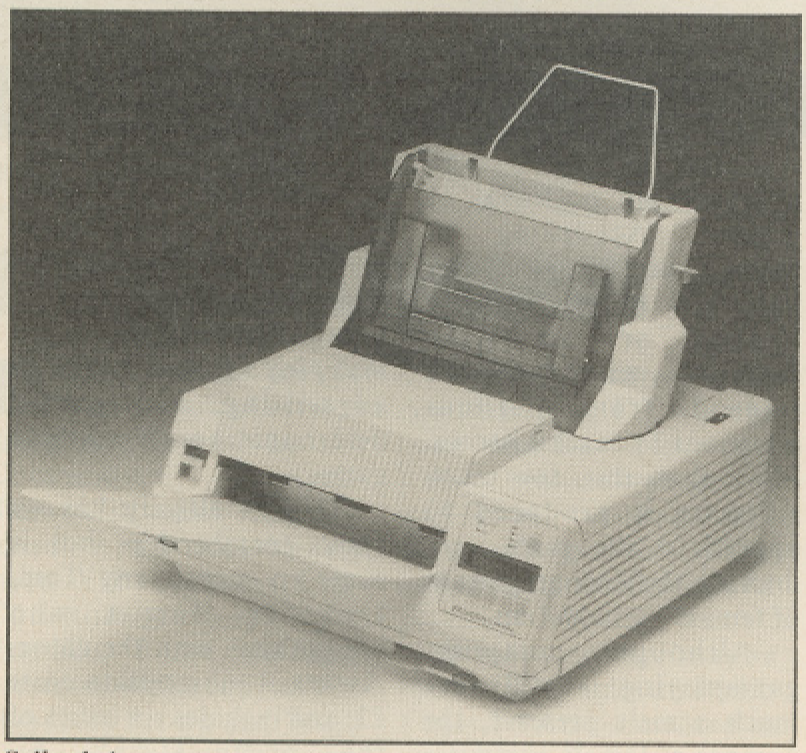
Following a £2.5 million investment, Seikosha UK plans a three year push to become one of the top printer manufacturers in Britain.

Seasoned Commodore owners may remember that the CBM VIC1525 printer was a thinly disguised Seikosha machine, but since those dim and distant days Seikosha has entered the new age of technology. Last year saw the release of the heavy duty SBP10 printer, which reached the heady speed of 800cps, making it the fastest dot matrix printer currently available.

1989 sees the start of the market assault with the launch of a LED printer, OP-105A, as an alternative to the Laser technology favoured by other companies. This will be supplemented by the release of two new printers later in the year.

The announcement coincides with the opening of the company's new factory in Hamburg, which follows the example set by Star and Epson to avoid the printer tariffs imposed by the European Community.

***Touchline:** Seikosha (UK) Ltd, Unit 14, Poyle 14, Newlands Drive, Colnbrook, Slough, Berkshire SL3 0DX.*



Seikosha's new, compact LED

Active Signings

Activision has signed two new labels and a development house as part of its continuing search for new talent. The development house, New Frontier Productions, is the latest venture for David Crosweller, who previously headed Infogrames in Britain. The deal promises a steady flow of games over the next two years, commencing in late summer.

The first of the new labels is a group of programmers who are all known to Activision through their mutual links with System 3. Vivid Images consists of *Last Ninja* and *Ikari Warriors* programmer John Twiddy, graphic artists Hugh Riley, and Mev Dinc, whose credits include the conversion of *Last Ninja 2* to Z80 format. The new label gives the team a more positive profile, and will hopefully bring them the recognition they deserve.

Motion Picture House is the second label, and will support individual authors and development houses. The label will also be used by the development team, which will be appointed to produce games for the Nintendo games machine.

Rod Cousens, Vice-President of Activision Europe, comments, "David has pulled together a very talented team, and we look forward to seeing an exceptional product line-up from New Frontier.

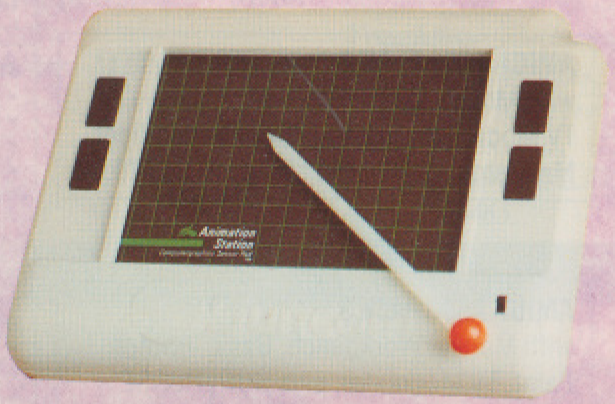
"The addition of Vivid Images Ltd and Motion Picture House to our group is further evidence of the great strides which we have made over the last year. We continue to attract and invest in new talent which, in turn, strengthens our position within the industry."

***Touchline:** Activision (UK) Ltd, Blake House, Manor Farm Road, Reading, Berks RG2 0JN. **Tel:** (0734) 311666.*

Graphics Guide

What are graphics all about, and what do you need to get the best from your computer?

By Kerry Fowler



A stunning graphic display is one of the most difficult effects to utilise effectively. This is especially true of the C64, but even the extended Basics of the C128, C16 and Plus 4 don't solve the problems totally. To get the best effects, commercial graphics programs are essential, and then the decision is which input device to use: keyboard, joystick, mouse, lightpen or graphics pad.

One of the strongest features of Commodore computers is their graphics capabilities. Sprites, characters and high resolution (hi-res), combined with a fairly wide range of colours, can add impact to games and utilities.

Hi-res graphics are generally supported by most of the available packages, but sprite designers and character defines are few and far between. This is true to such an extent that when a well-known programmer, Tony Crowther, published his own sprite and character defining routines in *Your Commodore* (3 in 1 Editor, August 86), the magazine had many enquiries from professional software houses who wished to use the program to design their own screens. This led to an updated version being created for our sister magazine, *Commodore Disk User* (3 in 1 Plus, Nov/Dec '87).

User-defined Characters

Character graphics involves the redefinition of the standard character set of the Commodore. The standard graphics are the letters and characters featured on the keyboard, and they are all based upon an 8x8 grid of pixels. A pixel is the smallest unit of a computer screen display – it's a single point of light which can be variously coloured to give the effect desired.

Each pixel on a grid can be switched on or off. For example, to produce the letter 'A', a character grid would take the formation shown in Figure 1. The problem is that a Commodore video chip can only access 16K blocks of memory, as shown in the sample C64 Bank in Table 1. This means that the screen, positioned at location 1024, can only grab its characters from locations 0 through to 16383. The computer circumvents this limitation by hardware trickery, but the rule is one which can never be broken by the programmer.

With the screen at 1024, the problem is that a Basic program uses the memory on either side of the screen – below the screen is the workspace for the ROM routine's variables and above it is Basic RAM for the program itself. This leaves very little space for any user defined graphics (UDGs), let alone sprites!

There are two solutions to this problem; the start of Basic memory can be moved up, or the screen itself can be placed in one of the other 16K blocks of memory. The authoritative guide to these methods is the relevant Programmer's Reference Guide from Commodore.

Mighty Sprites

Sprites are like small independent screens which can only display three extended characters measuring 8x21 pixels. If a sprite pixel is not turned on, the pixel becomes transparent and any characters behind the sprite will show through, but if the pixel is turned on, the sprite masks the background detail.

For some reason, Commodore decided not to implement sprites on the C16/Plus 4, which probably helped to add to the machines' downfall, because many of the programs written for the C64 involved sprites and couldn't easily be transferred to the newer computers.

Characters and sprites can be displayed as a single coloured sprite or multicoloured. Single-coloured or standard mode means that all of the turned on pixels are displayed in the same selected foreground colour, though this can vary from one character to the next. Multicolour is a slight misnoma, but it does allow three colours to be used.

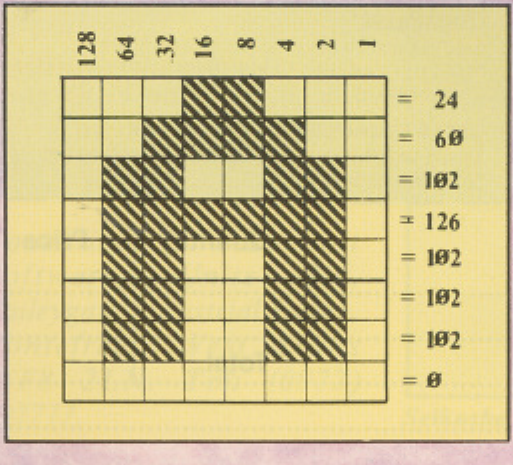
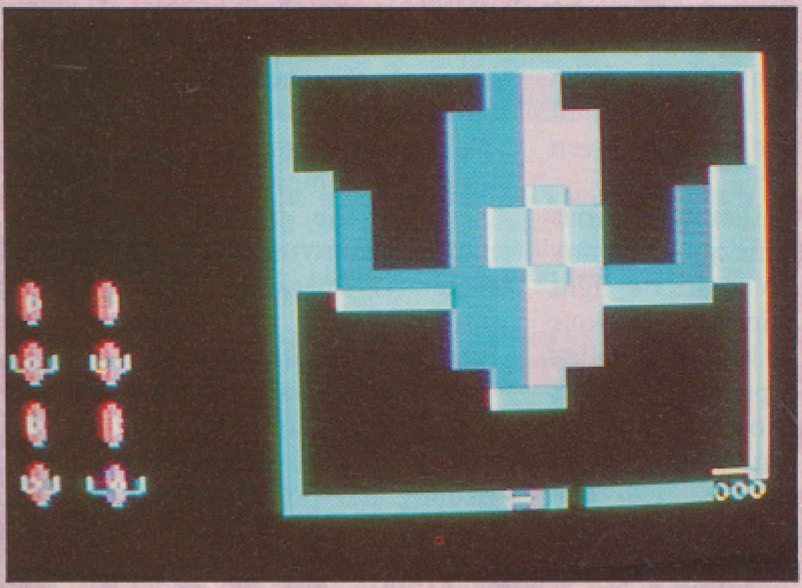
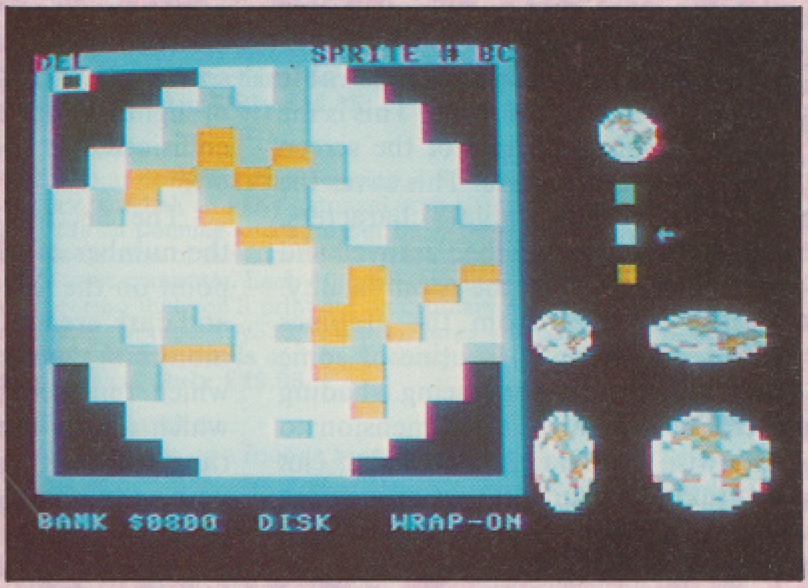


TABLE 1:

Table 1: C64 Banks for the VIC

Decimal	Hex	Bank Number
49152-65535	C000-FFFF	3
32768-49151	8000-BFFF	2
16384-32767	4000-7FFF	1
0-16383	0000-3FFF	0





As with standard mode, the foreground colour can be different for each individual character or sprite, but is the same throughout that particular character. The two extra colours are the same for all of the characters on the screen. Although sprites follow the same rule, their extra colours are taken from different memory locations, which means that the two ancilliary colours may be the same for all of the sprites, but can differ from the background characters.

The only problem with multicolour is that character definition falls to 4x8 pixels and 12x21 pixels for sprites. This means that the fine detail is not as good as for standard mode, and multicolour characters always have a jagged-edged appearance.

Higher Resolution?

High resolution is a misleading name, because the total number of screen pixels is the same as in standard low-resolution mode. The difference lies in the freedom which hi-res gives the user.

Hi-res graphics work on a similar principal to character graphics, and this is an important point to bear in mind when designing a screen display in this mode. Up to 256 UDGs can be defined in low resolution. At best, these would cover about a quarter of the screen if they were laid end to end, row by row. Hi-res allows a screenful of 1000 different characters to be defined.

The equivalent situation using UDGs, would mean reducing the screen to a 16x16 redefinable grid of characters, because each of the 256 characters can only be defined once. This gives a pixel resolution of 128x128 instead of the full screen resolution of 640x200 pixels – hence the justification of calling it *high-resolution*.

Each character on a hi-res screen has a fixed position, but the rules of UDG defining applies. This is especially notable in multicolour mode, because each 8x8 character area can only hold a maximum of three colours. The extra colour facilities of hi-res does mean that neighbouring characters can have three different multicolours, but the 8x8 pixel grid governs what can be done.

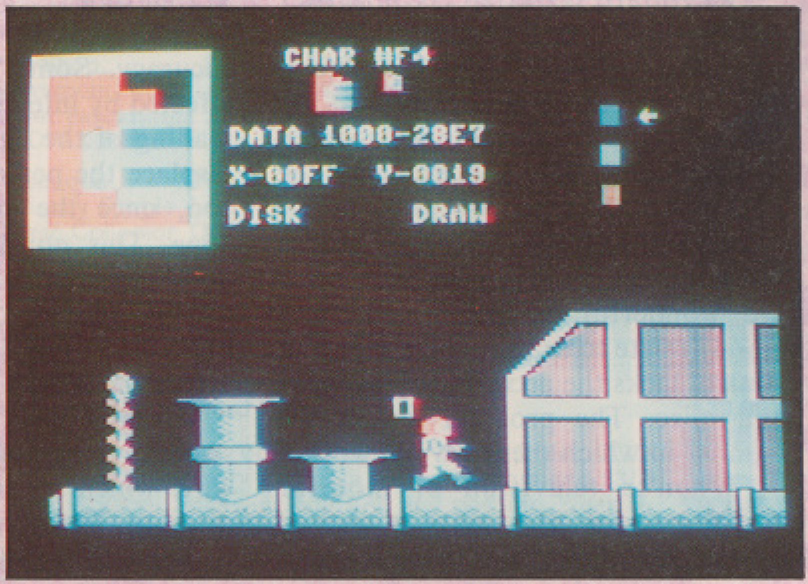
Pack Selection

There have been many graphics packs for designing hi-res images, but I don't know of any currently available packages for sprite or character definition which offer good value for money.

Trilogic's Expert cartridge system does have a sprite designer included in the package, but if this is the only facility which is of interest, then the price of £29.99 is a rather high one to pay. The best course of action is to check *Your Commodore*, and to look for the numerous sprite and character defining programs which crop up from time to time.

Hi-res packages are shown in the Touchline at the end of this article and typical features would include freehand drawing, filled and outlined shapes (principally circles and rectangles), single and connected lines, text mode, spray fill, solid fill, a range





of brush sizes, and zoom for closing in to add fine detail.

There are additional desirable features such as *windowing*. This is the ability to select an area of the screen and repeat it elsewhere. This saves the need to redraw similar characters, because it can be 'grabbed', moved and then modified using the zoom facility. To get the best from the graphics screen, a pattern fill routine of some kind can create convincing shading effects to add an extra dimension to otherwise 2D images. Patterning can also take the drudgery out of creating repetitive shapes such as the bricks in a wall or wallpaper.

A vital feature is usually called UNDO. This is most desirable when a FILL command goes wrong. Without UNDO, the image could be ruined beyond repair, with UNDO a press of a button can restore the image to its original state.

To fully utilise an image created by a graphics package, the user needs to know where the various elements are stored. Since this is rarely in the normal screen position (1024), it is also necessary to understand screen banking, so a good graphics book is essential. For hi-res, the Programmer's Reference Guide is not very useful, but there are many guides available in good bookshops – just check the index for a reference to banking, screen moving or something similar.

Graphic Aids

For most people, the cheapest tool for graphics is the joystick because it can be used for graphics or gameplaying. My own preference is a mouse, because it can be more easily operated with one hand. Even if this only means that the other hand is free to hold a cup of coffee, I find the use of a mouse more relaxing.

Keyboard control is another consideration, and sometimes offers the most accurate control system. Unless you are a touch typist, the problem is that it is tedious to watch the keyboard with one eye, and the screen with the other.

Several packages offer lightpens as input devices. In my experience, these are wonderful devices which most closely relate to drawing on paper, but control lacks the accuracy of joysticks and mice. The problem lies in the manner in which these devices work.

When a television or monitor draws a screen, the image is created by a stream of electrons being scanned across the lines of a screen. When an electron hits the phosphor coated screen energy is dissipated in the form of light. The greater the number of collisions, the brighter the light given off.

The electron gun therefore varies the number of electrons aimed at any point on the screen to create the light and dark areas of the image. In colour monitors, there are at least three guns which represent the primary colours which can be mixed to create the full range of hues and tints which makes up a colour image.

Each gun starts at the top left of the screen, and scans across the top line. Then it flies back to the left side of the next line and scans the next row. This continues until the bottom line is reached and completed, then the guns are all redirected back to the top left of the screen, and the process starts again.

Each point on the phosphor screen acts like a red-hot poker which has just been removed from the fire, it fades. The fade actually only takes a very tiny fraction of a second, and the first dot may be extinguished before the scan reaches the bottom line, but the whole process is so rapid that the eye cannot perceive this.

A computer initiates the screen scan, and can calculate where the guns are pointing at any particular time. If a light sensor is placed in front of the screen, it can calculate when the phosphor area in front of it is excited by registering the light that bursts forth. The computer can then calculate the time delay between triggering the screen scan, and the light sensor detecting the effect of the guns passing a particular spot. This allows a pixel in the computer's memory to be selected and operated on, according to the dictates of the software running at that time.

The problem is in getting pixel accuracy. Some software is severely affected by interrupts which can delay reading of the lightpen sufficiently to displace the perceived position of the pen, once the calculation has been done. This normally manifests itself as a twitching of the cursor on the display screen, or a line becoming broken or kinked as the pen is moved across the screen.

Graphic pads work on a different system. Imagine a crisscross grid of wires which are embedded in a plastic medium. The warp and weft of the grid is separated by a fraction of a millimetre, so that slight pressure will connect the two wires. The net effect is one of a thousand tiny switches which correspond to each pixel on the screen. This is analogous to the situation inside a graphics pad.

Once the signal is generated, internal electronics calculate which switch is operating, and sends the information to the computer, which then acts accordingly. The problem with a pad is that sometimes the point of the stylus is equidistant between two switches, and both are triggered. Usually the internal electronics will make an executive decision but sometimes one of the switches may constantly make and break and the screen cursor dances back and forth in sympathy.

On the subject of pads, most software is produced for the *Koala Pad*, which is still available in the States, but try getting one over here! It's impossible. Microprose market Suncom's misleadingly named *Animation Station*, which looks similar to the Koala but is totally incompatible with Koala software (based on experiments with Rainbird's OCP Studio). Fortunately, it does come with its own software (a pad-only version of *Blazing Paddles*), which is fairly comprehensive and the package is highly recommended for those who would like to try a graphics pad.

Whichever device is used, graphics are the root of all successful programs. Text is fine and may be essential but an illustrated manuscript is far more impressive than a paperback!

Touchline:
Blazing Paddles *£12.99 Datel Electronics with lightpen £24.99*
Graphics Support Utilities Disk *£12.99 Datel Electronics*
The Advanced OCP Art Studio *£24.95 Rainbird Software*
Artist 64 *£29.90 Wigmore House*
GEOS (Geowrite) *Disk only £24.95 Evesham Micros (free with Oceanic disk drive)*
NEOS Mouse *£24.95 Dimension Computers*
Datel Mouse *£24.99 Datel Electronics*
Animation Station *£ Suncom/ Microprose*

You may have noticed that these are all for the C64. GEOS incorporating GEOpaint is available for the C128 at £32.95, but there appears to be nothing for the C16/ Plus4.

F.S.S.L

FINANCIAL · SYSTEMS · SOFTWARE · LIMITED

▷ Award Maker Plus — NEW

Honour and thank people with more than just kind words or a letter—give them an award they'll keep for ever. With Award Maker Plus you create professional quality certificates with coloured or black, detailed borders, specialized graphics, and personalized messages, ready for framing.

Award Maker Plus is a software tool that lets you print awards, certificates, licenses, coupons, diplomas or other documents of your design. The program will print hundreds of predesigned Award Styles, allowing you to enter text, select the border style and colour, and in some styles even enter your own Hi-res pictures. Also, PrintShop graphics may be used as borders. The program comes complete with professional quality border designs, text fonts and graphic art. **Only £24.95.**

▷ 1764 RAM Expansion Unit — NEW

The Commodore 1764 RAM Expansion Unit adds 256K of storage capacity to the 64, four times the internal RAM, and comes with a replacement power supply to power both the computer and unit. RAMDOS software is supplied free to allow commercial programs, written for the unit to use it as a second, lightning speed drive. The 1764 Unit is ideal for programmers, GEOS users and users who need a second drive. **Only £99.95**

▷ Paper Clip Publisher — NEW

If you'd like to create eye-catching layouts, but don't enjoy playing with scissors and glue, Paper Clip Publisher has the solution. This powerful page layout program for the Commodore 64 lets you combine text from your favourite word processor and graphics from your favourite drawing program to create almost any layout you can think up.

To place an article into your layout, simply draw a box and pour in the text. Or draw several boxes and link them together, so the article is in several columns, or even on several pages. If you want the text to stand out, choose one of five fonts in any of four sizes. Then, give it some style—you can make it bold, italic, outlined, shadowed, reversed, or even upside down! Graphics are always a great way to get a readers attention PaperClip Publisher lets you use clip art from Doodle, Newsroom, Print Shop, Printmaster, and Stop Press. You can also create simple graphics in PaperClip Publisher's Graphics Editor. Once you've selected a graphic, simply pour it into your layout, then enlarge or reduce it to fit your specific needs.

PaperClip Publisher is a powerful layout tool but, more important, it is also easy to use. Your tools are displayed right on your screen—you simply point and click with your joystick or mouse. Other powerful commands are available in pull-down menus, so you don't have to go searching through your manual to find a command. Whether you are creating a single-page advertisement or a multiple-page newsletter, PaperClip Publisher will help you get the job done quickly and professionally. **Only £34.95.**

▷ BillBoard Maker — NEW

Are you restricted by the normal one page graphics? Would you like to be able to print posters four feet by three feet? Now you can, using BillBoard Maker, store the graphic to disk from any of the popular drawing packages and then load it back with BillBoard Maker. The optimizer will enlarge the display and also remove any jagged edges and then you can flip, crop, cut, mirror, strip colour, negative or frame the new image and then print it to your printer. **Only £24.95.**

▷ Photo Finish — NEW

This is a NLQ program, Near Laser Quality. On your dot matrix printer you can printout graphics with quality approaching that of a laser printer. It uses an optimizer which automatically computes the amount of available space between diagonal pixels and fills them in with additional pixels, before it prints out your picture. As a result your picture is drawn with smooth curves and diagonals rather than jagged edges. You can take almost any Hi-res picture and produce 640×400 resolution on almost any dot-matrix printer. This program is easily tested, print a picture and look at the curves made up of triangles, Photo Finish would round those edges. **Only £24.95.**

▷ 64 PRODUCTS ▷

▷ Project Planner

Too much to manage at once? Where do you begin? Need some help to organize your work? Could you be more efficient? With Project Planner you and your micro can cut any problem or task down to size, isolate key tasks, and anticipate problems you had not foreseen, judge the importance of conflicting priorities, and recognize critical elements in any plan, work out how long the project will take and where money and time can be saved, display your plan as a clear chart, then watch the effects when you modify and edit your plan. **Only £14.95.**

▷ ToyShop — NEW

A balloon-powered steam engine, a medieval catapult, an accurate sundial, an early carousel with horses, a clockwork bank that runs on pennies and a rubber band powered helicopter are just a selection of the 20 models which can be produced with the aid of your computer. Each of these toys is a marvel of engineering that you'll print out on ordinary computer paper and assemble yourself. ToyShop even includes a supply of card, wire, balloons, rubber bands and wood along with a comprehensive, 210 page manual. **Only £29.95.**

▷ Fun Biorythms — NEW

Print your own or a friends biorythms cycle in a chart form to printer. The cycle shows your physical, emotional and intellectual days over a set period. **Only £4.95.**

▷ Super Cat

Super Cat allows you to catalogue your disk collection. You can catalogue up to 640 disks and 5000 titles per catalogue, then at the press of a key, search and find the location of that missing program. A variety of reports can be printed including programs in alphabetical order, programs by disk, three types of disk labels and more. Super Cat will sort 1000 programs in just 8 seconds. **Only £19.95.**

▷ Super Aide — NEW

All purpose utility program for the 64. Super Aide hides in memory in a safe, out of the way place, yet is immediately accessible at all times. Super Aide provides, Bi-directional screen scrolling, auto line numbering, line deletion, renumber, trace, a monitor, disassembler, disk directory, display a file to screen, change this to that, find, unnew and more. **Only £12.95.**

▷ 64 Doctor — NEW

A professional diagnostic cartridge for the 64. It automatically tests and reports to screen the condition of the keyboard, serial and user ports, kernal ROM, video display chips, all video banks, NMI and IRQ interrupts, BASIC, CIA and sound chips. 64 Doctor comes with a free comprehensive fault and diagnosis and repair guide to allow you to repair the computer. **Only £18.95.**

▷ 1541 Repair and Maintenance Guide — NEW

The repair and maintenance book unveils the electronics and mechanics of the 1541 disk drive. The concise presentation of the material and introduction to digital electronics assures that even if you have no previous experience in electronics you can still repair your disk drive. Topics covered include general maintenance, speed adjustments, lubrication techniques, differences in drive versions, testing, adjust read/write head and more. An excellent guide, and useful when aligning your drive. Large A4 format. **Only £12.95.**

●GEOS 64	£24.95	GEOS 128	£32.95
DeskPack Plus 64/128	£21.95	FontPack Plus 64/128	£16.95
GEOSpell	£16.95	GEOPublish	£32.95
GEOProgrammer	£32.95	GEOCalc 128	£32.95
GEOFile 128	£32.95	Becker Basic	£29.95
Official Programmers Guide	£7.95	Anatomy of the 64	£4.95
Anatomy of the 1541	£4.95	Tricks and Tips 64	£4.95
Graphics on the 64	£4.95	Ideas for your 64	£4.95
Machine Language	£4.95	Peeks and Pokes	£4.95
PrintShop	£29.95	800 ClipArts	£19.95
Newsroom	£24.95	Super C	£29.95
Super Pascal	£14.95	Cobol 64	£29.95
CADPAK 64	£24.95	Chart Pack 64	£24.95

Technique	Tutorial for programming graphics	£4.95
Basically Simple	Learn how to program Basic	£4.95
Comic Collector	Organize your collection of comics	£4.95
EasySign	Learn to use the sign language	£4.95
Label Maker	Print over 10 different labels	£19.95
PPM 64	Look after and follow shares	£39.95
Pocket Planner	Spreadsheet with sideways	£39.95
Fontmaster 64	Word Processor with extra fonts	£34.95
Pocket Filer	Database with Report Writer	£39.95
Pocket Writer	Word Processor	£39.95

●Order with Access, VISA, Cheque or Postal Order, COD and Express delivery available, foreign orders add £2.50 per package. Prices include VAT and postage. Please allow 14 days for delivery.

NEW *Utilities and Educational catalogue available – call (0386) 553153*

18 High Street · Pershore · Worcs · WR10 1BG · Tel: (0386) 553153

▽ 64 PRODUCTS ▽ 64 PRODUCTS ▽ 64 PRODUCTS ▽ 64 PRODUCTS ▽ 64 PRODUCTS ▽



If you don't have access to a wordprocessor, but want to send a printed letter to someone, 'Letter Writer' can help

by A. Wright

Letter Writer

Have you ever wanted to send someone a short letter that looked efficient? I'm sure many of us have sent one of those grovelling, pretty-please-type letters at some time or another. Well, the letter will look best if it's typewritten – after all, if your handwriting looks as though its been penned by a spider that spent the previous evening in a whisky bottle, you're not going to get very far, are you.

This is where 'Letter Writer' comes to the rescue. It's not designed to be a wordprocessor program, it simply allows you to compose your letter quickly and then print out the finished masterpiece. One feature I'm sure users of 40 column wordprocessor programs will appreciate is the fact that the display is in 80 columns, and thus you can see each line on screen as it would appear on paper.

Getting It All In

Firstly, it's important to note that you *must* have a copy of the "ROM 80" program by Jens Meyer, which gives the C64 an 80-column screen display. This appeared in the January 1988 issue of *Your Commodore.* Also, note that both programs can be entered with or without the "ROM 80" program resident in memory.

Type in Letter Writer and SAVE it to tape or disk, then type in Letter Printer, and SAVE it immediately after. I find it best to have the programs stored immediately after each other in the following order: ROM 80, Letter Writer, Letter Printer.

Using the program

If you haven't already done so, LOAD and RUN the ROM 80 program. Next, LOAD and RUN Letter Writer 2. If all is well, you'll see the following: at the top of the screen is a copyright message; below this, on the left, is a row and column count. Taking up the centre of the screen is a large box; this is the 'window' though which you will see your text. At the bottom is the message 'PRESS "CTRL" FOR HELP SCREEN'.

Type something (apart from "CTRL") – you can enter anything except Commodore keyboard graphics, and the left and up-arrow keys. Notice that if you enter quotes, they don't appear, but an up-arrow does. this is because printing quotes can have a weird effect. Don't worry! The arrows are exchanged for quotes just before printing.

N.B.: when you enter text, the program has to check a lot of things before it PRINTs each character, and these checks slow the program down a little. As a result, if you are a fast typist, you may notice that the screen display doesn't quite keep up with you. This shouldn't be a problem, as the characters are stored in the keyboard buffer. However, the buffer only holds ten characters, so if you type fast, some characters may be missed. The only solution, I'm afraid, is to slow down!

Right: on with the show... pressing RETURN, or reaching the end of a line, will move the cursor to the beginning of the next line. Note that if there is any text on the next line, the cursor will move to the end of that text. Pressing DELETE will erase the character on the left of the cursor. If you are in the first column of a row, pressing DELETE will move the cursor to the line above, erasing the last character on that line. If you wish to move to the line above without deleting the last character, press the up-arrow key. The cursor will move to the correct position on the above line. If you move up or down on to a line which already contains 60 characters, the cursor will be in column 61, which is not normally possible. In

this situation, you cannot enter a character. You can only move up or down.

Now, when you've reached the bottom of the box, and you fill the line or press RETURN, the text will scroll up by one line. When you are on the top line, and press the up-arrow key or delete past the beginning of the line, the text will scroll down by one line. Note that you may only have up to 60 lines, and you can't scroll up past line 1 or down past line 60. I chose 60 lines as this is as many as can fit neatly onto an sheet of A4 paper.

Being BASIC, the scroll routines are quite slow. If you find it's too slow for you, I suggest the following:

When you reach the bottom line, and have entered the text for that line, press RETURN a few times (10 at most). Then, when the scrolling has finished, move the cursor up to the correct position and continue typing.

THE OPTIONS

Now that you know how to enter your text, it's time to learn what you can do with it. The first thing to do is press "CTRL" to access the help screen. You *must* go to the help screen before you can use any of the options.

The top of the screen contains simple reminders about how to enter your text – the options are at the bottom. To select, simply enter the number which is to the left of the option you require. The options are:

1. RETURN TO TEXT
This option does not affect the text in any way. It simply returns you to where you were before calling the help screen. It is the only option which doesn't ask "Are you sure?" before continuing.

2. GO TO LINE
Who'd like to guess what this does? After making sure you're sure, simply enter the number (1-60) of the line you wish to go to. You will be returned to your text, with the cursor at the correct position on the line you selected.

3. SET TEXT NAME
This option allows you to enter a filename which is used when saving or loading your text. When you first RUN the program, or if you enter a blank line, the current name is given as "No name!" i.e. the program will save or look for an un-named file, *not* one with the name "No name!" To give a filename, type in the name and press RETURN. The name stays the same unless you enter another, or select "DELETE SHEET" (see option 6).

4. MOVE TEXT
On selecting this option, you are asked for the start and end lines of the block you wish to move, and the new start line of the block. When you've entered these, the block will be moved, and you will be returned to your text at the line you were on before calling the help screen. You should be aware of the following things when using this option:

(a) You CAN move a block to start within itself, i.e. if you move text from lines 1-20, you can define the new start lines as line 10.
(b) The block to be moved must fit, i.e. you can't move a block of 20 lines to start at line 50. If you try this, the block will not be moved; you will get the message "Block will not fit." and will be returned to your text where you left it.
(c) To move only one line, enter the same number for both the start and end lines.
(d) Most importantly, moving a block will over-write any text that is in the new position. Also, the lines are deleted from their old position.

5. COPY TEXT
This works in exactly the same way as "MOVE TEXT", except that the lines are not deleted from their old position.

6. DELETE TEXT
On selecting this option, you will be asked whether you'd like to clear a block, or the entire 'sheet'. Selecting 'block' will prompt for the start and end lines of the block to be cleared. The block will then be cleared, and you will be returned to your text at the correct position. If you select 'sheet', the program is re-RUN. This clears all text and resets the filename to "No name!"

7. SAVE TEXT
I'm sure there's no need to explain this option, but I will tell you that the current filename is displayed along with the "Are you sure?" prompt. Therefore, if you wish to change the name before saving, just press "N" and go to the "SET TEXT NAME" option. When the text has saved, you'll be put back where you were.

8. LOAD TEXT
Again, there's no need to explain this. Note, however, that when text has loaded, the cursor will be on line 1 of the text. If you can't remember the name of the text you wish to load, go to the "SET TEXT NAME" option and enter a null string (press RETURN). The program will then load the first file it comes to. Also note that when a file is loaded in this way, the name is NOT set to that of the file, i.e. it is left blank (or "No name!")

9. QUIT PROG
This is self-explanatory.

PRINTING OUT YOUR TEXT

I had to write the printing program as a separate unit, as my printer is non-Commodore, and I have to use an interface; sadly, my interface software uses the same area of memory as ROM 80. (For those who are interested, my printer is an Epson MX-80 F/T Type III, and my interface is the "Commodore Connexion", which I bought in Boots for £19.95 about two years ago. It may be cheap, but it suits my needs perfectly.)

My printer defaults to the English character font on power-up, and while in this font, printing a hash (" ") will result in a pound sign being printed. So, if I want to print a pound-sign, I send a hash to the printer, hence line 9 of the Letter Printer 2 program. If I want to print a hash, I must change to the American character font, send a hash, and change back to the English font, hence line 11.

Confused? You should be! Let me make it simple: if your printer can print a pound-sign AND a hash from the SAME font, delete lines 9 and 11 from the Letter Printer 2 program. By the way, a list of the printer control codes used can be found at the end of this article.

Right. When you've saved your text using the Letter Writer 2 program, switch off your computer, connect your printer, and switch them both on. Load your interface software (if necessary), and then LOAD and RUN the Letter Printer 2 program. Follow

TABLE 1: PRINTER CONTROL CODES

CONTROL CODE	LINE	ACTION
CHR$(27)+"R"+CHR$(0)	11	Set printer to American character set.
CHR$(27)+"R"+CHR$(3)	11	Set printer to English character set.
CHR$(27)"@"	15	Initialise printer.
CHR$(27)"8"	15	Disable paper-end detector.
CHR$(27)"3"CHR$(18)	15	Set line spacing to 18/216 inch.

the on-screen instructions. After a short wait (while the 'up-arrows' are being exchanged for quotes), your text will be printed, with a nice even border all round. (At least, it will if you've got your paper in straight!).

That's about it, I think. I hope you find the program useful. It may not be up to the standard of 'Easy Script' and other word-processors, but, as far as I know, it's only the third true 80-column (well, almost) "word-processor" available to those without a disk drive! (YC Writer, from the *Your Commodore Serious Users Guide*, and Letter Writer 1 being the other two.)

There are four functions which were added after the above instructions were written. To use these functions, you DO NOT have to go to the help screen, just press the required key(s). The functions are:

1. Holding down either SHIFT key and pressing CLR/HOME will clear the current line of text and place the cursor at the beginning of the line. This saves time when clearing a line because you've noticed a mistake.
2. Pressing f1 will centre the line of text that the cursor is currently on, thus saving time when entering headings etc.
3. Pressing f3 will place the end of the current line at the far right. Please note that this is NOT a right-hand justification, as the lines do not remain parallel to the left margin. As an example of how this can be used, type in your address, with each line starting at the left margin. Then, move the cursor up to the first line of the address and press f3. By doing this for each line of the address, you do not have to play around entering spaces to line up your address on the right hand side.
4. Pressing f5 will place the current line so that it starts at the leftmost position (the opposite of f3). This was just included in case you press f3 on the wrong line, or any other similar mistakes.

If you use functions 2-4 on a line with leading or trailing spaces, they may take a few seconds to work, as these spaces must be 'trimmed off' for the functions to work correctly.

Right. That *is* it. Have fun! And who knows? There may be a Letter Writer 3 on the way! It won't be for a while though, 'coz I've got a big blister from typing out these blasted instructions. I'm off for a cuppa. Till next time...

THUNDER BLADE

One of the hottest coin-op licenses of the year, and the sequel to Sega's *Afterburner*, *Thunder Blade* is now competing head to head with that high-flyer for the coverted Christmas number-one spot.

The game puts you behind the wheel of the Thunder Blade, an advanced attack helicopter that's armour plated, crash resistant and armed with a 1,500 round 30mm cannon and air-to-ground missiles. Unfortunately, in the C64 version it's hard to find any difference between these two projectiles, and they seem to appear randomly on the screen destroying anything and everything.

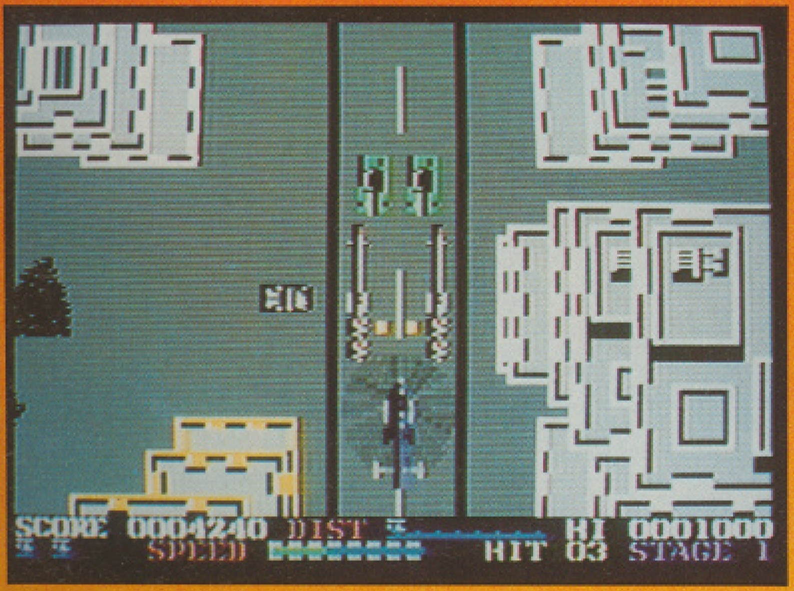
Your mission is to destroy the advancing forces of an evil dictator who is threatening your land. This gives you the opportunity to blast everything in sight as you battle your way through the game's four levels. Each level consists of both top-down and 3D-view combat sequences in which you shoot it out with enemy attack helicopters, Bengal Tiger tanks, Barracuda torpedo cruisers, Corsair fighter aircraft, armoured cars and *Burner* aircraft (US Gold obviously hopes you'll enjoy blowing them out of the skies). The end of each level is guarded by an aptly named super-fortress with enough firepower to down a squadron of Thunder Blades.

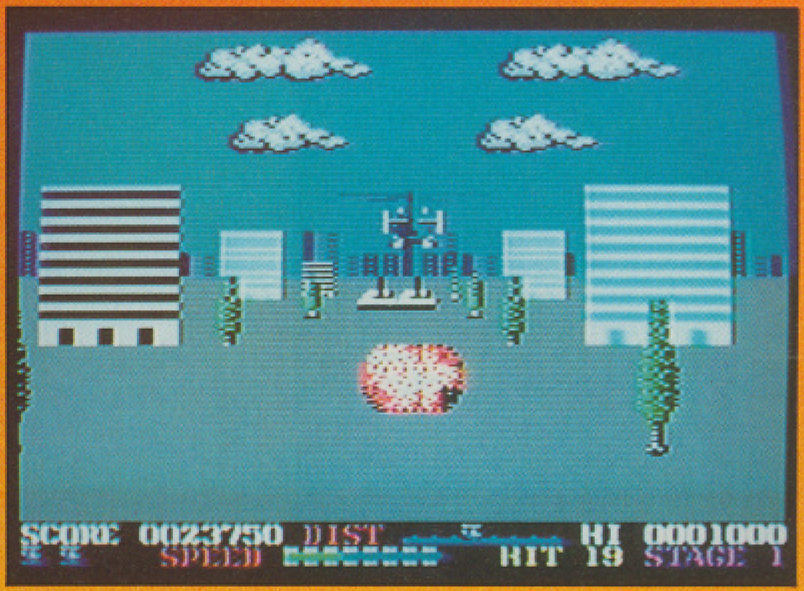
The first level is set in the concrete jungle of Skyscraper City, and begins with a top-down view as you take-off from the Pepsi Cola heli-pad. You're soon busy dodging fireballs blasted at you from tanks, armoured cars and even the building themselves. The graphics at this point are very disappointing – the buildings are just four stacked squares that move uneasily over each other to give you the bad impression of a tall tower block. While I appreciate that programmers need to take shortcuts to keep the speed of the game to arcade level this just isn't up to the high standards we have come to expect.

Having said that, the gameplay will drive you back for more and more. Once you've cleared the last car and silenced the tanks, the view switches to a 3D perspective, and you have to run the gauntlet in Skyscraper city once again. This time the enemy helicopters are more effective, and you get your first chance to scrap some *Afterburner*-style aircraft.

If you survive that you win the chance to tackle the first super fortress, a giant warship bristling with guns and missile launchers that will test your skills and Thunder Blade's manoeuvrability as you hover back and forth dismantling the defences. A bar at the bottom of the screen displays important data such as your score, hits, the high score, your speed and how far you've managed to get through the stage.

Stage two takes you out of the city and into the mountains and deserts. The battle remains the same, but now stone columns replace the skyscrapers. At the end of this level, a massive troop carrier forms the super fortress, and presents even greater challenge.





Stages three and four take you down the river delta to face the Barracuda torpedo cruisers, and onto the refinery to deal with a flying fortress and the command super-frotress – the ultimate test for advanced pilots.

Thunder Blade was a top arcade game, and is almost guaranteed great success as a coin-op conversion, but of all the available versions, the C64 one seems to be the worst. Although, the game play is good, the graphics are poor, missiles and cannons have the same effect, and you get only three Thunder Blades a game, whereas the other versions get five. It's good, but it could have been so much better.

Touchline:
Title: *Thunder Blade.* ***Supplier:*** *US Gold, Units 2/3 Holford Way, Holford, Birmingham B6 7AX.* ***Tel:*** *021 356 3388.* ***Price:*** *£9.99.*

THE ULTIMATE BACKUP CARTRIDGE GOES ONE BETTER...

EVEN MORE POWERFUL, MORE FR...

TOTAL BACKUP POWER

The most powerfull backup system ever devised. Unstoppable freezer system. Freeze at any point. Specially designed to work with game software. Just press the magic button to backup even the most heavily protected programs.

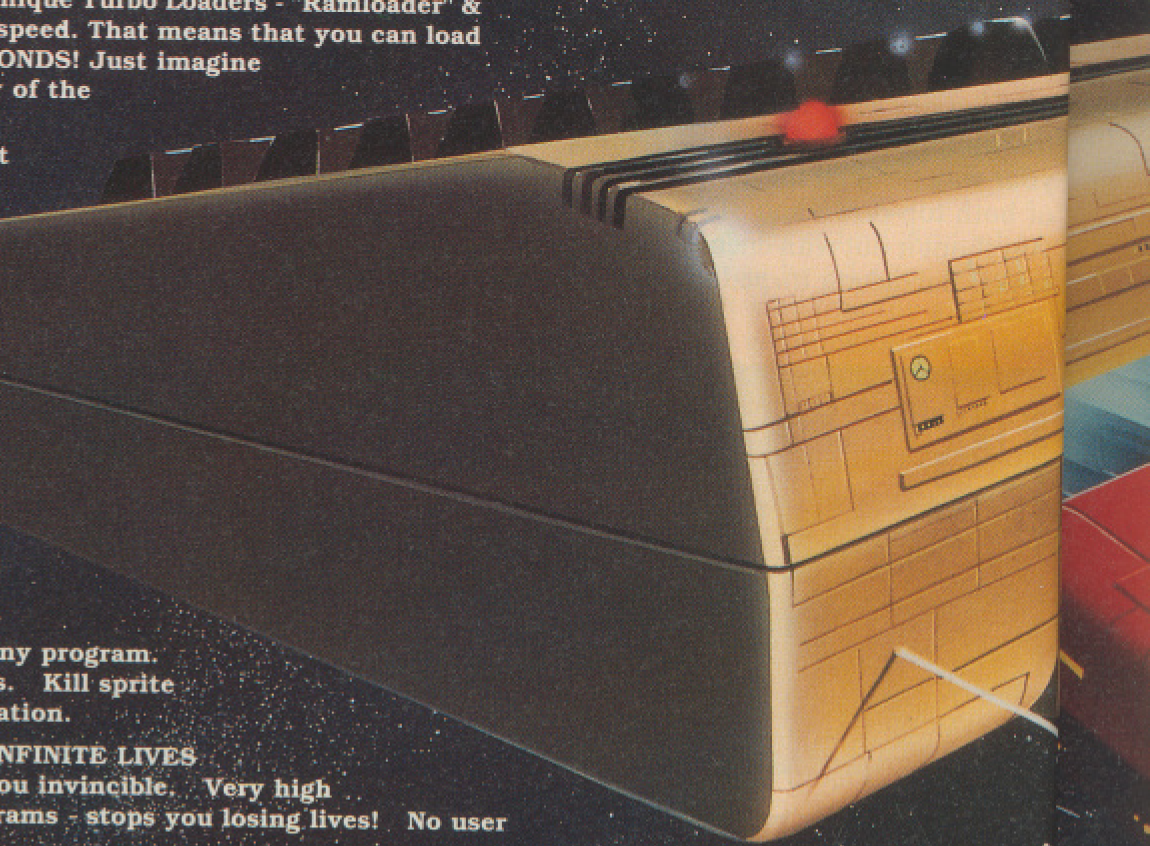
❑ TAPE TO TAPE ❑ ❑ TAPE TO DISK ❑
❑ DISK TO DISK ❑ ❑ DISK TO TAPE ❑

VERY fast & efficient program compaction. Single part save at TURBO speed to disk or tape. Backups turboload INDEPENDANTLY of the cartridge. No "programing" or user knowledge required.

THE PROCESS IS AUTOMATIC - JUST GIVE THE BACKUP A NAME.

ONLY £34.99 POST FREE

JUST LOOK AT THESE FEATURES...



- **TURBO RELOAD.** Action Replay Mk V has 2 unique Turbo Loaders - "Ramloader" & "Warp 25". Both work at up to 25 times normal speed. That means that you can load a typical program (200 blocks) in around 6 SECONDS! Just imagine your backups loading completely independantly of the cartridge in seconds instead of minutes.
- **PRINTER DUMP.** Freeze any game and print out the screen. Eg. loading picture, high score screen, etc. Works with most printers. MPS 801, 803, Star, Epson, etc. Double size, 16 shades, reverse print option. Very versatile - no user knowledge required.
- **PICTURE SAVE.** Save any Hires multi colour screen to disk at the push of a button. Compatible with Blazing Paddles, Koala, Artist 64, Image System, etc.
- **SPRITE CONTROL.** Freeze the action - view animations. Load, save and modify sprites in any program. Flip, reverse, delete etc. Customise your games. Kill sprite collisions - make yourself invincible. 64K operation.
- **POKEFINDER GENERAL.** AN AUTOMATIC INFINITE LIVES GENERATOR! Finds those pokes which make you invincible. Very high success rate - works with both old and new programs - stops you losing lives! No user knowledge required. Great fun!
- **MULTISTAGE TRANSFER.** Even transfers multistage programs from tape to disk. The extra parts fast load - a unique feature. Enhancement diskis required for multi-loaders (see below).
- **SUPER COMPACTOR.** Ultra efficient program compaction techniques. Each program saved as a single file. 3 programs per disk side - 6 programs per disk, if you use both sides.
- **TEXT SCREEN EDITOR.** Modify the text screen on a frozen program. Customise your games by adding your name to the title screen - change colours, etc., then make a backup. Also a great programmers aid.

  Verify, relocated save. Fast format - 12 seconds. Built in file copy - works with long files. Built in 1541 disk copy - 1 or 2 drives. Read directory, send disk commands. Change disk name, device number. Load direct - no need type filename.
- **SUPERFAST DISK OPERATION.** Load 200 blocks in just SIX SECONDS. Works with any program of any length. Works with multiload programs. Versatile - Backups, Basic, Monitor. Works with all drives including 1581. Use both sides of disk (1571). Standard format - no file conversion required. Superfast Save,

GRAPHICS SUPPORT UTILITIES DISK

A range of utilities to make the most of your high res pictures created with popular graphics utilities or captured with Action Replays unique picture grabber.

SLIDE SHOW. View your favourite screens in a slide show type display. Move from screen to screen - keyboard or joystick control. Very easy to use.

BLOW UP. A unique utility to allow you to take any part of your picture & 'blow it up' to full screen size. Even fills the border with powerful sprite handling techniques. Very easy to use - simple commands. An interesting utility.

SPRITE EDITOR. A complete sprite editor helps you create or edit sprites. Full colour display. Animate to view movements. Action Replay can capture/insert sprites with any program - this editor is a perfect companion.

MESSAGE MAKER. Takes your favourite screen - created with a graphics package or captured with Action Replay & turns it into a scrolling screen message complete with music. Send screens to your friends with music & text scrolling. Simple text editor - easy to use. Choice of music. An exiting utility. Finished screens stand alone. ONLY £12.99

WHAT THE REVIEWERS SAID

"I'm stunned, amazed and totally impressed. This is easily the best value for money cartridge. *The Cartridge King!*"
Commodore Disk User

ACTION REPLAY ENHANCEMENT DISK

The biggest and best collection of special parameters and file copy programs for transferring non-standard multi-load tapes to disk - games like LAST NINJA, CALIFORNIA GAMES, LEADERBOARD, DRAGON'S LAIR - NINETY titles in all. Almost all major titles covered. Latest edition includes SALAMANDER, HAWKEYE, THE GAMES series, STREETFIGHTER, VINDICATOR, ALIEN SYNDROME, PLATOON and many more. Just about every major multiload title can be transferred fully to disk. Cheats for infinite time, lives etc.

The GRAPHIC SLIDESHOW - latest edition displays multicolour pictures or loading screens saved by Action Replay or any major Art Package - Blazing Paddles, Koala, Advanced Art Studio, Artist 64 etc. Lots of fun.

Only £8.50. Upgrades - send £3.50 plus old disk.

N

ACTION REPLAY Mk V

FCBM64/128

PROFESSIONAL

HAS ARRIVED

FRIDLY & NOW EVEN MORE FEATURES!

ALL FEATURES AVAILABLE TO TAPE OR DISK USERS.

ACTION REPLAY MK V differs from all other cartridges because it combines an 8K RAM with a FULL 32K operating system ROM. That means that ALL features are INSTANTLY AVAILABLE AT ALL TIMES.

WARNING!! Other systems use outdated technology which severly limits performance. Action Replay's state of the art hardware gives you MORE POWER, MORE SPEED, MORE FACILITIES than any other cartridge. There really is no comparison. Here are just some of the features...

LY
.99
FREE



- **MORE TAPE FACILITIES.** Dual speed tape turbo for backups. Very fast, very reliable. Programmers tape turbo - very compatible. Even works with sequential files. Built in slideshow for hires pictures. You don't need a disk drive to use Action Replay.
- **TOOLKIT COMMANDS.** Auto line numbering. Defined function keys. Old, Delete, Merge, Append, Linesave. Plist - list directory direct to printer. Single stroke load, save, directory. Switchable disk/tape turbo.
- **FULLY INTEGRATED OPERATION.** The MK V 'Professional' has an onboard custom LSI LOGIC PROCESSING CHIP that integrates the whole range of utilities and makes them available at the press of a button at any time.
- **CENTRONICS INTERFACE.** For parallel printers, eg Epson, Star, prints listings with graphic characters. Send escape codes - make full use of your printer's extra facilities. Auto detect of parallel printer - no special commands required.
- **PROFESSIONAL MACHINE CODE MONITOR.** Full 64k monitor available at all times. Examine all memory, registers, IO, stack of any frozen program. Full range of commands, plus the luxuries that only a high capacity RAM/ROM system can offer: Assemble, disassemble, Hex/Ascii. Interpret in Ascii or screen codes. Fill, Hunt, Compare, Transfer memory. Number conversion, Register, Go. Load, Save, Verify (turbo, tape or disk). Two way scrolling of all screen displays. Output to printer (CBM or Centronics). Directory, error channel, 2 drive operation. Disk Monitor - read block, write block, assemble/disassemble drive memory etc. Hex calculator - add, subtract, multiply, divide. Unique set break/set freeze system. JSR Freeze. Full "floating" operation - corrupts no memory. Call Monitor from Basic or Freezer. Call Freezer from any point in your program.

AR MkV
Reloads an average BACK-UP in
6 Seconds!!

o the title

r 2 drives.

orks with
sk (1571).

t value for

ISK
programs
AST NINJA,
tles in all.
ANDER,
LIEN SYN-
tiload title
s etc.
ctures or
- Blazing
fun.

UPGRADE INFORMATION

MK IV Professional to Mk V Professional - just send £9.99 & we will send you a new Mk V Operating System Chip. No need to send your old cartridge - just plug in the new chip!

Mk IV (Standard) to Mk V Professional - send your old cartridge plus £15.99 & we will upgrade it to Mk V Professional. (allow 14 days).

PERFORMANCE PROMISE

Action Replay will backup any program that any other cartridge can backup - and more! It also has an unmatched range of onboard features. Before you buy, check our competitors ads to see what they offer and see how many of the Action Replay MK V features are either not there or have to be loaded from seperate disks etc. When you buy Action Replay, if you don't find our claims to be true, then return it within 14 days for a full refund.

ALL ORDERS NORMALLY DESPATCHED WITHIN 48 HRS

HOW TO ORDER ...

BY PHONE	BY POST	FAX
0782 744707 24hr Credit Card Line	Send cheques/POs made payable to "Datel Electronics"	0782 744292 UK ORDERS POST FREE EUROPE ADD £1 OVERSEAS ADD £3

PRICES AND SPECIFICATIONS CORRECT AT TIME OF PRESS AND SUBJECT TO CHANGE WITHOUT NOTICE

CALLERS WELCOME - Please reserve goods by telephone prior to visit.

DATEL ELECTRONICS

DATEL ELECTRONICS LTD., FENTON INDUSTRIAL ESTATE GOVAN ROAD, FENTON, STOKE-ON-TRENT, ENGLAND.

SALES ONLY	TECHNICAL ONLY
0782 744707	0782 744324

Microprose Soccer





At a time when nearly everyone who ever strapped on a pair of boots has licensed a football game Microprose launches it's first sports simulation and makes a departure from the usual diet of combat flying. It's literally a game of two halves, as the flip side of the disk contains a different version of the game. On one side you can succeed where others have failed and bring home the world cup, or flip the disk and try the American six-a-side version of the game.

Whichever game you choose you'll be playing one of the fastest and best arcade-style football games, with a screen display showing a scrolling window over a top-down view of a match. You can adjust the length of each game up to 12 minutes, but two minutes will be long enough until you get used to the controls. You'd be surprised how many goals you can let in in the time, particularly if you're playing against Brazil.

The 11-a-side game contains 29 teams in all, ranging from Oman and New Zealand to Argentina, Italy and Brazil, and the first challenge you should try is to play up the order of the teams, moving up every time you win, but down when you lose. This'll give you time to learn the basic moves – you'll have to, as not even Bobby Robson can stand humiliating defeats for too long.

Soon your players will be sliding in and tackling you'll be able to control throw ins and corners, and you'll be able to decide quickly whether to bring your keeper out to narrow down the angle, or wait on the line to dive to save the shot, instead of dithering somewhere in between. Eventually, you can even select the type of shot to try, including a straightforward volley, a banana shot, chip shot or Pele-style overhead kick. Whatever the style, you'll be able to enjoy it again as you see an action replay of all goals.

Microprose Soccer is the first football game to incorporate the effects of weather, if only in a limited way as without warning a downpour can begin, and thunder-and-lightning can liven up the dullest 0-0 draws. This adds a new dimension to sliding tackles, as the player can carry on sliding and even spin out of control, leaving the opposing player with free space to work in.

A world cup tournament is played in six groups, with the first two teams in each qualifying for a second knockout stage. Each group has a team from each of the 'seeded groups', that include teams according to their ability and past record. For example Brazil, West Germany and Holland are in the first seed group, and England are in the second (which seems quite generous on recent form). Therefore you can adjust your chances of success in qualifying by picking the team you will represent – if you choose to play Brazil, then you'll have an easier group than if you opted for Cameroon, but will probably come unstuck in the knockout stage.

If you flip the disk, you can swap Italy and Argentina for Houston and Miami as you compete in the American-style indoor league, which by a strange coincidence is organised in six groups with the first two progressing to the next stage, etc, etc...

Microprose Soccer was programmed by Sensible Software, and it has plenty of those touches that make a good game great, and the professional feel of a Microprose game. Some people will dislike the top-down perspective as its not a camera angle we're used to, but perhaps it's the only realistic way of getting 22 players on a football field.

Touchline:
Title: *Microprose Soccer.* ***Supplier:*** *Microprose, 2, Market Place, Tetbury, Gloucs., GL8 8DA.* ***Tel:*** *0666 54326.* ***Price:*** *£14.95 cass. £19.95 disk.*

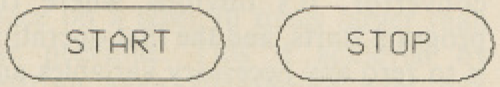
A Flow of Ideas

Formal flowcharts are the best way to show how a program works

By Norman Doyle

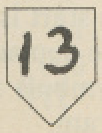
Any book on programming will tell you that REM statements can be used to remind the programmer how a program was originally conceived. Having gone back to modify many of my older programs, I have found that REMs only work to a certain degree. The best answer is to create a formal flowchart, because this can aid conversion from one machine to another, as well as later updating.

A flowchart is a graphic display of the logic held within a program and, as the old adage says, a picture is worth a thousand words. Formal flowcharts, or flow diagrams as they are also known, rely on a series of box symbols which represent certain specific actions within the program. All flowcharts should start and end with a *terminal:*

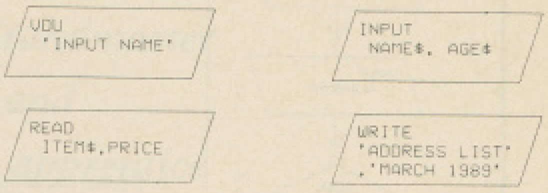


To indicate the start, the terminal has START written inside it, and the end terminal has STOP in it – isn't logic wonderful.

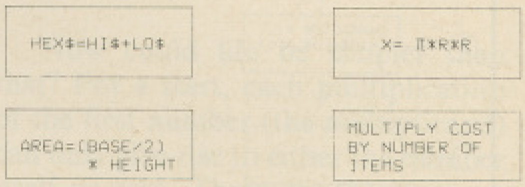
Some flowcharts spread across several sheets of paper, so a special *offpage connector* is used which contains the number of the page to which it connects. Similarly, the routine on the page to which it refers starts with an offpage connector containing the number of the page from which it continues.



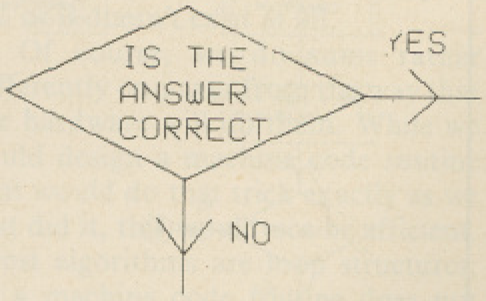
Probably the most used symbol in a program is the *input/output* parallelogram (I/O symbol). This contains such things as print statements, input requests, DATA reads, printout commands and disk or tape accesses. The Data manipulations



are contained in rectangular *processing* symbols. The contents include the handling of strings and logic operations, as well as the display of more conventional mathematical formulas:

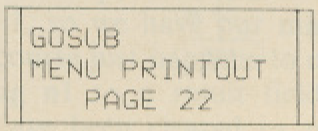


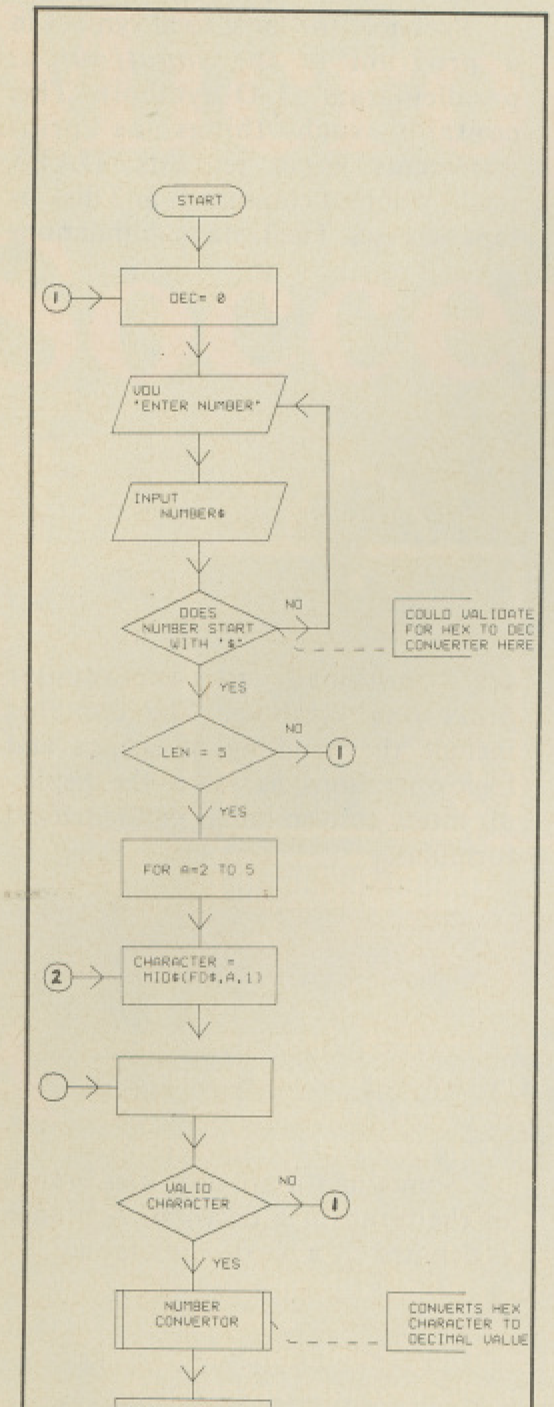
Sometimes a program can branch as the result of a decision. A branch is denoted by a *decision* diamond:



Notice the arrows on the branch lines. These are *flow direction* indicators, and should appear between every box to indicate the direction the program is taking, because the direction of flow may not always be obvious.

Subroutines would be difficult to show on a single diagram, so a special form of the processing rectangle has been devised to cope with this. The *predefined process* symbol looks like this:





Symbols contain all of the program information, but there may be occasions when extra information is necessary. This equates to the REM statement in Basic. The *annotation* symbol is an open-ended box which connects to the flowchart by a dotted line to the box to which it refers:



A typical flowchart looks like the one in Diagram 1. Notice that the branch necessitates the drawing of a long line which connects the decision symbol back to the re-entry point. In a program with several of these links, the page would soon look very untidy, so a *connector* symbol has been devised which is only used to connect two parts of a flowchart contained on a single page. If a branch goes over to a new page then the offpage connector is used.

This is the range of symbols commonly used in flowcharting. Professional diagrams also use another set of symbols for various storage devices and communications links, but these would rarely be needed in a small home computing environment.

Exploded Diagram

Diagram 1 shows a hex to decimal convertor. It's obvious where the program starts, and the first operation is to zero any necessary variables and then print a request for a number input onto the screen. Although Commodore Basics allow INPUT statements to print the text and the input prompt on the same line, this may not be true in all cases, so the input of a number string is listed as a separate action.

The input is then tested to see if it starts with a dollar sign. If it does, the program continues, otherwise it returns for a proper input. At this point the program could include a decimal to hex convertor in a future development. Although the full branch line back to the start is shown, in this case it could be replaced by the use of connector symbols as used on the rest of the diagram.

Following through the conversion, the program next checks if the string is five characters long (dollar plus for significant characters) – if this is not the case then the program loops back to position 3. If true, the convertion process is started with a loop which takes each character in turn, excluding the dollar sign, by using a MID$ command.

Each character is tested for validity, and if any one fails this test the program loops back for a new input. Valid characters are passed to a subroutine called *number convertor*, which has an annotation symbol attached to explain what the subroutine does.

The subroutine actually converts the hex character to its decimal equivalent and returns it as the variable VALUE. This variable is then added to variable DECIMAL, and then multiplied by 16 to prepare it for the next hex place.

The loop is checked to see if all values have been taken and, if not, the next value of A is taken. If the loop has ended, the program prints the message 'DECIMAL =' and then the decimal value. Note the ways of expressing text and variable values. Text is enclosed in single quotes and variables as a plain name.

The program closes by asking if another input is required, and loops back if this is the case. Otherwide, the program ends at the stop symbol. Converting this to a program is simple. Try it for yourself and then check the listings pages to see how I did it.

```
PROGRAM: HEX/DEC CONVERTOR

30   10 D=0
81   20 INPUT"ENTER NUMBER IN HEX
        ";N$
0F   30 IF LEFT$(N$,1)<>"$"THEN10
DC   40 IF LEN(N$)<>5THEN10
11   50 FORA=2TO5
91   60 C=ASC(MID$(N$,A,1))
7F   70 IFC>47ANDC<59THEN100
66   80 IFC>64ANDC<71THEN100
34   90 GOTO10
52   100 GOSUB180
01   110 D=(D+V):IFA<5THEND=D*16
FA   120 NEXT
1E   130 PRINT"DECIMAL=";D
D9   140 PRINT"[DOWN]ANOTHER INPU
        T"
0D   150 GETA$:IFA$="Y"THEN10
51   160 IFA$<>"N"THEN150
2A   170 END
26   180 V=C-48:IFC<59THEN200
40   190 V=V-7
46   200 RETURN
```

The Software Spellbook

In which Wizard Emeritus Myron Patch, Grandmaster of the Honourable Order of Coders and Algorithmicists, offers essential lore for the apprentice Numeromancer

To use a C64 without a few essential spells (the superstitous refer to these as 'software') is as pointless as trying to invoke the Moon Goddess without spilling the blood of a virgin ram. Therefore in this series I offer, to those who have the ears that can take in such wisdom, a few small fragments of forbidden love that may dramatically increase your ability in the arcane branch of knowledge known as machine coding.

I assume that everyone out there knows the fundamental Coder Level ' spells such as Addition and Sub-'raction. Without these, of course, no progress to the more fearsome level 2 Multiplication and Division is possible. It is the first of these that we shall be looking at.

The easiest form of Multiplication spell to understand is shown in listing 1, with a flowchart This multiplies two single-byte numbers NUM1 and NUM2 to give a two-byte result – RESULT. Already, this shows us something essential about Multiplication. You must always, when multiplying a number of n1 bytes by another n2 long, allow for a result n1*n2 long. This is a general principle of software magick, which the unwary ignore at their peril.

So how does the spell work? All apprentice grade spellworkers already have a founding in the popular, but vulgar magical language known as decimal, its founding being the number 10. You may be surprised to find that performing multiplication spells in this language is actually more complicated than using a method based on the magical law of 2, which we usually call binary.

Here is a typical base ten multiply:

```
          23
         x45
         ---
5x23-    105
 CARRY:    1
        ----
         115
  4x23-   82
  CARRY:   1
        ----
         035
 CARRY:    1
        ----
        1035
```

The same essential components of this are used in binary. You can see that there is a process of repeated addition, but that, because of the numeric system, each addition takes place on a level 10 times larger than the previous. This is achieved by shifting each succeeding sum one position to the left. Secondly, note the use of the carry. This, of course, is use when a multiply goes beyond the range of one column (i.e. greater than 9).

All this is obvious, but my reason for explaining it should become clear when we repeat the same process for two small binary numbers:

```
             111

            x101
            ----
111x1:       111
111x0:      000
111x1:     111
           ----
           00011
CARRY:     111
           ----
          111011
```

Now could life be simpler than that? For a start, each multiplication of the first number (the multiplicand) can only give rise to either the number itself or zero. There are therefore no carries in the multiplication, since the range of one column is 0-1 and you cannot get two no matter how you try. The carries come in in the addition, but once again, a carry can only be a 1 or it doesn't exist at all.

Of course, we do sums rather differently on paper from the way that the hardware will do them. While we could design a machine code routine that would do that trick exactly as we just did it, this would not be efficient. Most algorithms are loop structures, so a machine code routine does not pile up a heap of multiply results and add them up at the end. It instead performs an addition to a cumulative total every time it goes once around the loop.

The process is like this for n1Xn2:

1. Is leftmost digit of n2 a 1? This is done by shifting it leftwards, and if it is then the carry flag will be set. If it is not then proceed to 3.
2. It's a 1, so we have our multiply. This of course just happens to consist of adding n1 to a result field (this should have been cleared to zero at the start of course). The addition may

Listing 1 + Listing 2

```
 10 ;
 20 ;multiply 1 byte x 1 byte
 30 ;
 40 num1       equ $fb
 50 num2       equ $fc
 60 result     equ $fd
 70 ;
 80 ;
 90 multiply   lda #0;          clear result lo
100            sta result+1;    and hi
110            ldx #$8;         set up counter
120 multloop   asl num2;        leftmost 1?
130            bcc multskip;    no
140            clc
150            adc num1;        add to lo
160            bcc multskip;    no carry
170            inc result+1;    add to hi
171 multskip   dex;             dec loop ctr
172            beq multend;     if zero exit
180            asla;            shift result lo
190            rol result+1;    & result hi
191            clc
192            bcc multloop;    go back
220 multend    sta result;      leave result lo
230            rts
```

```
 10 ;
 20 ;multiply 2 bytes by one
 30 ;
 40 num1       equ $fb
 50 num2       equ $fd
 60 result     equ $2a7
 70 ;
 80 ;
 90 mult2      lda #0;          clear result
100            sta result+1;    fields
110            sta result+2
120            ldx #16;         set up counter
130 mult2loop  asl num1;        leftmost 1?
140            rol num1+1
150            bcc mult2skip;   no
160            clc;             yes so add
170            adc num2;        multiplier
180            bcc mult2skip;   to the
190            inc result+1;    three
200            bne mult2skip;   bytes
210            inc result+2
220 mult2skip  dex;             decrement loop
230            beq mult2end;    if zero exit
240            asla;            shift result
250            rol result+1;    fields
260            rol result+2
270            clc;             return to
280            bcc mult2loop;   start of loop
290 mult2end   sta result;      store result lo
300            rts
```

be more or less complicated, depending on how many bytes you are multiplying, but it's always there.

3. The result field must now be shifted one to the left ready for the next placing.

4. proceed to 1

Fine, but we only want to do the process a certain number of times. To achieve this, a machine code routine will use one of the registers as a loop counter. The number of times you need to go around the loop depends on the number of bits in n2, as you may guess. For a single-byte number this will be eight, for a two-byte sixteen, and so forth.

Besides the simple single-byte multiply, I have also included a listing for a routine that will multiply a 2-byte number by 1 byte to give a 3-byte result. The principle is exactly the same, so I leave you to figure this out yourself.

Quick and Dirty

What we have just described is a general-purpose multiply. Even non-adepts know, however, that when in a hurry, there are often faster methods. When performing the awesome Ritual of Matrix Multiplication, many a newcromancer has substituted an ordinary lizard for the fearsome Death Reptile from the Jungle of Balak-Nizaram and survived (of course those that survive are the only ones who can tell us about it).

Quick and dirty multiplications are excellent things when a particular result is required. You need to be quite clear when they are acceptable, though. Bear in mind that when you shift a number one bit to the left, you multiply it by two, as any power-of-two multiplication, if that is exactly what you want, can be achieved this way, and is extremely fast.

Other numbers can be broken down into powers of two and multiplied rapidly in this way. The commonest example is ten. Multiplying by ten is exactly the same as doing two multiplies, one by eight, and one by two, and adding the result. The last routine does exactly that. Look out for when you need to use routines like these. Speed is of the essence in real-time graphic spells, so always consider a quick and dirty method if you can, rather than a general-purpose routine which will always go eight, 16 or more times around a loop, no matter what size of number is present.

Calling all wizards

In issues to come I will be illustrating ways to do many other common spells. The next page of the Spell book will deal with division.

Illustrative spells, however, are not always the fastest. I will be interested in hearing from any adepts out there who have designed algorithms of their own and can explain them. There can simply be faster ways of doing common things. If you can pare microseconds off a standard random-number generator, for example, I want to hear from you.

If we select your algorithm, it may well win you a crisp five-pound note. So send us an assembler source code listing plus a plain-language explanation of the magickal proceedings and a small measure of fame and fortune may be yours.

Listing 3

```
 10 ;
 20 ;multiply 1 byte x 10
 30 ;
 40 num1       equ $fb
 50 result     equ $fd
 60 ;
 70 ;
 80 multx10    lda #0
 90            sta num1+1
100            asl num1
110            rol num1+1
120            lda num1+1
130            sta result+1
140            lda num1
150            asla
160            rol result+1
170            asla
180            rol result+1
190            clc
200            adc num1
210            sta result
220            lda num1+1
230            adc result+1
240            sta result+1
250 multend    rts
```





Pac-mania

Unless you happen to be a hermit living in a cave in the unpopular end of Outer Mongolia, you will know and either love or hate *Pacman*. That yellow mouth that munches dots while the machine eats your money at an alarming rate has already starred in two computer games, the original coin-op conversion and the game of the arcade sequel, *Pacland*, which was a sideways scrolling arcade adventure featuring ghosts in planes and other equally unlikely events. Now, *Pacman*'s back in 3D!

Pac-maniacs can now munch their way through 3D scrolling screens filled with dots to eat, ghosts to avoid, and power pills to turn the ghosts pale and into bonus points, as they scuttle back to their base to recharge. As in the original game, fruit appears for a limited time to tempt you into the middle for a juicy bonus, but also into the range of patrolling ghosts and away from the power pills.

Pac-mania is more than just a 3D version of the original game – although that would probably be enough for most Pac-maniacs as it adds a few extra features that will make the game a challenge for all players whatever their skills. From the opening screen you can select which level you will start at, and either start slowly with the easier screens or jump straight into the thick of the action.

The game consists of four different screens that you visit again and again as the things speed up and generally get tougher, that go under the misleading names of Block Town, Pacman's Park, Sandbo Land and Jungly Steps.

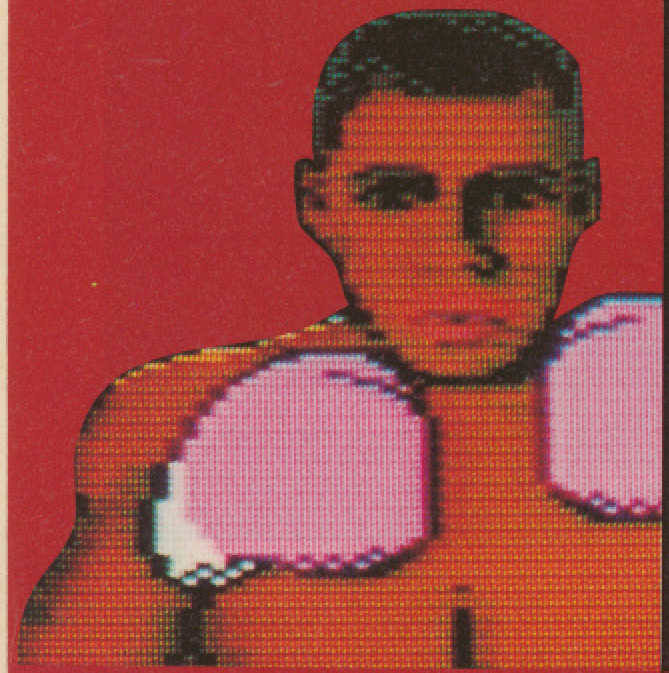
Clyde, the chief ghost, has recruited two new spooks, Sue and Flunky, to pursue you, but Pacman has a new trick up his sleeve – pressing the fire button makes him jump, over ghosts if necessary, to escape from tight corners. Pacman also has the help of two new powerpills that appear in place of some fruit. A red powerpill gives you double points, which can really help to rack up those high scores, and a green pill that temporarily turbo charges Pacman so that he hurtles around the maze.

Pacman has always enjoyed a cult status, and has been played by those who don't normally play computer games (such as *Your Commodore* production people), and with the addition of 3D graphics, this following should grow and grow.

Touchline:
Title: *Pac-mania.* ***Supplier:*** *Grandslam Entertainments, 12-18 Paul Street, London EC2A 4JS.* ***Price:*** *£9.95*

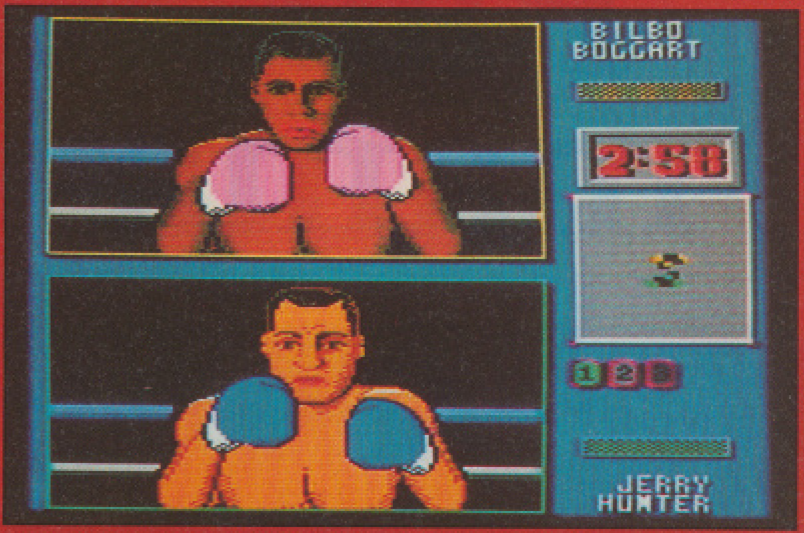








T.K.O



There have been many attempts at a boxing simulation in the past, some good, some dreadful. However Accolade, a company that's had considerable success in the past with its sports games such as *Hardball* and, *True to Form*, has come up with something a bit different this time.

T.K.O. stands for technical knockout, the term applied when the referee watches your face being reshaped into an unrecognisable pulp, and decides it's time to throw in your towel for you. The object of the game is to become champion of the world, and apart from the current champ, there are four main contenders selected randomly from a pool of 12. Alternatively, you can choose instead to slug it out with a friend. After every fight, the statistics of each fighter are updated and saved to disk.

You can tweak the characteristics of your boxer to your heart's content, altering his stance and adjusting the power between left and right-hand, head and body punches, strength or speed. Finally, your boxer has a built-in weakness, and it's up to you to decide whether you want him built to last but likely to cut up around the face, or vice versa.

Where this game differs from other boxing simulations is in the actual fight itself. Instead of showing the whole ring, you get a head-on view of your opponent. As punches are thrown, you can see them land or miss, and judge your tactics accordingly. Likewise, thanks to the split screen, you can see what punishment your man is taking.

There are five different guard positions that you can adopt, protecting your head, chin, throat, chest and stomach. Your offensive punches are determined by your guard – high guard, ample will lead to jabbing, and you can hook from a central position, or uppercut from a low guard. Where you aim is determined by moving the joystick to the desired position and pressing fire. Targets range from the solar plexus through body, jaw, mouth, nose and eye.

At the end of each round the scoreboard is displayed, showing how many punches you have thrown, how many hit, what damage they caused and so on. The trick is not to throw a lot of punches, but to try and make the ones that you do throw count. One good punch is a lot more effective than a dozen light slaps.

Although this game looks good and plays well, there's not enough variety to satisfy any but the keenest fight fans.

Touchline:
Title: *T.K.O.* ***Supplier:*** *Electronic Arts, 11-49 Station Road, Langley, Berks, SL3 8XN.* ***Price:*** *£9.95 (cass) £14.95 (disk)*



Risk

If your knowledge of geography is limited to such places as Irkutsk, Yakutsk, Kamchatka and the Congo, then it's a dead cert. that at some stage or other, you've played the board game *Risk*. Now, Leisure Genius have signed up the rights from Parker Brothers, and produced a computerised version of this classic board game.

Risk does not concern itself with building pretty little houses on Mayfair or trying to make the longest word possible from a random assortment of letters. No, the object of this game is world domination, no less.

The world is divided up into 42 regions spread over six different continents, and the aim is to have your armies in all of them. At the beginning of each turn, you receive a number of armies according to how many territories you own, with bonuses if you occupy a complete continent. You can deploy these armies as you see fit. Only adjacent territories can be attacked, the result being calculated by rolling dice, and you can never leave anywhere unguarded.

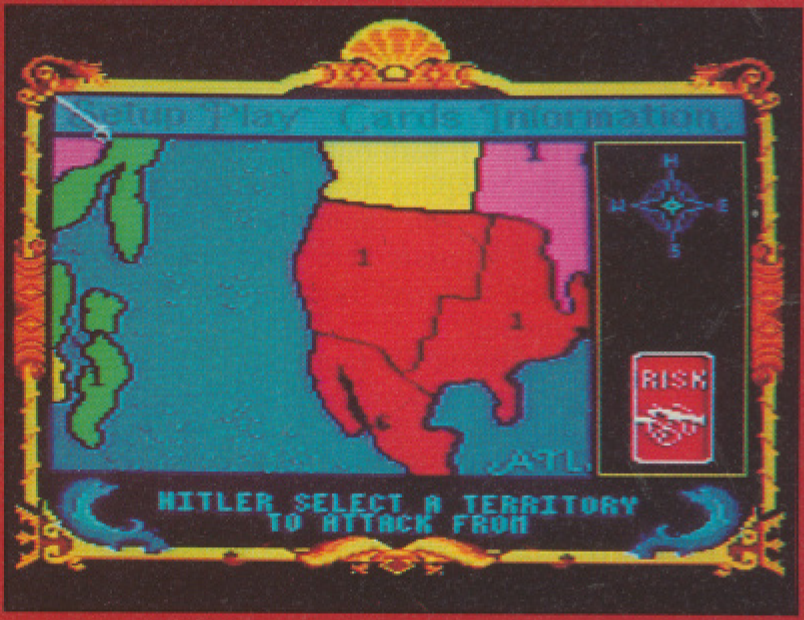
If you capture a territory in a turn, you receive a card. These go to make to sets, which can be traded in at a later turn for additional forces. From one to six players can play, with any number being controlled by the computer, so that you are never stuck for an opponent. In addition, each computer player can play on one of three different skill levels so that you set up exactly the sort of challenge that you fancy.

One of the problems with the board game is that everybody who you played against seemed to have a slightly different set of rules! This is because the game evolved with time, and there were several differences between the American and British versions. This is no longer a problem – through a series of playing options, you can now select whatever parameters you choose. The main differences involve setting up the board initially, and calculating the number of armies due to you whenever it's your turn to play. In addition, you choose from long and short games, different victory conditions, and may save any half completed game for a later date.

Control of the game is simplicity itself. The map can be scrolled in small increments by using a compass, or in large chunks by moving the pointer to the edge of the screen. Choices are made from pull-down menus. The graphics are large, clear and colourful, and there is no trouble differentiating between who owns what and how many armies there are in each country.

As a rule, I have not been particularly enchanted with computer versions of board games, but this conversion of *Risk* works admirably. This, I am sure, is partly because the game itself works well as a one player strategy game, whereas other board games such as *Monopoly* don't. I would not like to play this version with five or six humans, preferring to spread out round the board instead, but that minor quibble apart, this is a superb version of the game and highly recommended to all would-be Genghis Khans.

Touchline:
Title: Risk ***Supplier:*** *Leisure Genius, 2-4 Vernon Yard, Portobello Road, London, W11 2DX.* ***Tel:*** *01-727 8070.* ***Price:*** *£14.95 (disk), £12.95 (cass)*





Operation Wolf

There has been a dearth of good shoot-em-ups this year, but now your trigger finger can stop itching. *Operation Wolf* has arrived! Far and away the most successful arcade game of recent times, the only thing that Ocean have not managed to include in their conversion is the Uzi sub-machine gun that sits in front of the arcade console.

Basically, you're on a one man mission of death and destruction. Sure, you have to rescue some prisoners towards the end of the game, but think of all the bodies you can pump full of lead en route. This is definitely a game for someone with the mentality of a Rambo rather than a General Patton.

The game comes in six horizontally scrolling stages. In each section, there is a detachment of forces that has to be taken out before you can progress. These consist not only of enemy soldiers, but also tanks, gunboats and helicopters, and it is for this reason that you are equipped with a handful of rockets, as well as clips of ammo for your Uzi.

Considering the nature of your mission, you go into battle decidedly ill-prepared, with only nine clips of ammo and five rockets. Extra weapons do appear on the screen which you can pick up if you shoot them, but there is frequently a conflict of interests here – do you take out the heavy enemy forces and risk running out of ammo, or risk life and limb instead?

Because of the paucity of ammunition, it's not clever to keep your finger permanently on the fire button. Instead, restrict yourself to short bursts. Longer bursts are required to dispose of the tanks and helicopters should you run out of rockets, but this is worthwhile as they do most damage to you. Other objects to look out for as you blast your way merrily round the screen are open boxes of dynamite. Shooting these will effectively destroy everything on screen, so bide your time before blowing it up. You can also gain unlimited fire power, but this lasts for a short time only.

As you progress through the levels, so the difficulty of your task increases – not only do you have more men to kill, but the accuracy of your aim needs to be spot-on as well. The large soldiers start wearing bullet-proof vests, and have to be shot through the head rather than the body.

Naturally, you're not indestructible, and an energy level shows your current state of health. You can improve your state of health by completing a level or shooting a power pill but still, the sliding scale moves inexorably towards your impending death. The best way to postpone your impending doom is by making sure that you take out the vehicles and paratroopes as quickly as possible. Also, shooting innocent civilians such as the nurses is considered to be a definite no-no, and you are severely penalised for each indiscretion.

Instead of the machine gun in the arcade version, aiming is achieved by manoeuvring a crosshair sight round the screen. Joystick response is excellent, once you've got the hang of what's going on. There is also the option to use a Neos mouse which is easier to use, although to compensate, you get less ammunition to start with.

The conversion to an eight bit machine is excellent, and all the addictive qualities of the arcade machine are present. Mindless violence it might be, but it is great fun as well. Strongly recommended.

Touchline:
Title: *Operation Wolf* ***Supplier:*** *Ocean* ***Price:*** *£9.95 (cass), £14.95 (disk).*





Extending Basic

In this second article on Extended Basic, we develop a COLOR command and GOTO and GOSUB variable

By Burghard-Henry Lehmann

The main disadvantage with Commodore BASIC is that it doesn't give the programmer any access to the better facilities of the C64. There isn't even a command that lets you change the screen colours!

In the first article of this series on extending the BASIC of the C64, we introduced a simple command that changes the border colour. We can now expand on this and introduce a COLOR command that let's you change the ink, the paper and the border colours, all with one command.

'COLOR'

For the command itself I've chosen the American spelling, which is 'color', because this means one less letter to type in and saves us few bytes in the machine code routine. Remember, the last two letters of the command will be tokenized by the computer, because OR is a Basic function which has the token hex B0 (decimal 176)! Because of this I compare in line 600 of four program with $B0 instead of '0' and 'R'.

The syntax of the COLOR command is COLOR a,b,c – whereby a is the value for the ink colour, b is the value for the paper colour and c is the value for the border colour. As you can see, the separator between each parameter is a comma, even though I haven't included an error check in the routine that forces the user to use a comma and only a comma. This might be a job you like to try your hand at. Here is a clue: to force the system to give a Syntax Error report, jump to $AF08. This will load the X-register with $0B and then print the 'Syntax Error' message.

As I promised in the last article, we're going to make the use of variables possible. In order to make this possible, we use two ROM-routines: $AD8A, which evaluates an expression, and $B7F7, which transfers a value from the floating point accumulator # 1 into the zero page variables $14/15. $AD8A gets an expression from the Basic text and evaluates it. The value of the expression (or "variable") is then put into what is called the "floating point accumulator" (short: "flp accu").

The Floating Point Accumulator

There are two floating point accumulators: flp accu # 1 ($61 – $65) and flp accu # 2 ($69 – $6E). These are used by the C64 for all the mathematical calculations it does. For example, if an addition is to be done, the fist number is put into flp accu # 1 and the second number is put into flp accu # 2. Then the addition itself is done. Finally, the result is left in flp accu # 1.

Each floating point value is stored in five bytes: the first byte represents the exponent of the number, the second byte holds the sign (+ or -) and mantissa 1, and the final three bytes hold mantissa $2 to 4. This makes it possible to hold floating point numbers of any size – negative as well as positive – and do calculations with them with a high degree of accuracy. But its also quite cumbersome and slow. That's why it's usually better to develop your own mathematical routines rather than use the calculator in ROM.

Nevertheless, in our routine it is convenient to use the ROM-routine because not only does $AD8A interpret a simple variable, but it can also do a mathematical expression. This enables us to enter, for example, COLOR INK+1, PAPER+B, BORDER*2/2. $B7F7 gets a value from flp accu # 1 and puts it, in integer form, which all we need, in zero page $14/15 and the Y-register (low byte)

and the acuumulator (high byte).

All we have to do then is load the contents of the Y-register into the system variables which set the colour – that is, 646 for the ink colour, 53201 for the paper colour and 53280 for the border colour.

GOTO with variables

The next two routines (lines 1060-1120 and 1170-1310) enable us to enter GOTO and GOSUB with a variable. That is, you can enter a line, like "SUBROUTINE=1000", and then "GOSUB SUBROUTINE". (But remember, as with all variables in Commodore Basic, only the first two letters of the variable name are taken into account!)

To modify GOTO so that a variable or expression is accepted, is really very simple: since the token for GOTO is hex 89, we compare in line 460 with that value, and, if it matches, make the program flow jump to our routine (lines 1060-1120).

First, we point to the first character of the parameters of the GOTO command by calling CHARGET (line 1080). The next two lines (1090-1100) do the same thing as we did with our COLOR command: the variable or expression is evalutated ($AD8A), and the value which results is put from flp accu # 1 into zero page $14/15 ($B7F7).

Finally, we call the ROM routine at $A8A3 which executes the GOTO command itself, that is, makes the flow of the Basic program jump forward or backward to the line specified after GOTO. This routine expects the target line number to be in zero page $14/15. That's why gathered the value from flp accu # 1 into those variables with the $B7F7 routine earlier on.

GOSUB with variables

GOSUM (lines 1170-1310) is a bit more complicated. First, the relevant information for the line to which the program flow has to return – when the RETURN command is given – has to be stored in the machine stack, and then an ordinary GOTO has to be performed.

Because of GOSUB, I had to modify the overall structure of our Extended Basics program slightly as well. You may have noticed that at the beginning of the main routine, three ROM routines are used: first, CHARGET is called, to get the token of the command itself or the first character of our extended command (line 380). Then our main routine is called (and not jumped to, as in the program in the previous article – line 390), and finally a jump is made to $A7AE (line 400), that is, the ROM routine which executes the next Basic statement.

In other words, our main routine (lines 440-1510) has been made wholly into a subroutine. I copied Lines 1180-1290 of the GOSUB routine from the ROM routine at $A883. I couldn't give a JSR to this

continued on page 87.

Basic Demo

```
10 BEGINNING=20 : DELAY=110
20 INK=INT(RND(1)*16)
30 PRINT "INK=";INK;" ";
40 PAPER=INT(RND(1)*16)
50 PRINT TAB(15) "PAPER=";PAPER;" ";
60 BRD=INT(RND(1)*16)
70 PRINT TAB(28) "BORDER=";BRD
80 COLOR INK,PAPER,BRD
90 GOSUB DELAY
100 GOTO BEGINNING
110 FOR N=1 TO 500
120 NEXT
130 RETURN
```

The Machine Code

```
10                ORG 49152
20                ENT
30    ;
40    CHARGET     EQU $0073
50    EXECVECT    EQU $0308
60    ;
70    ;
80    ;
90    ;TURN EXTENDED BASIC ON
100   ;BY CHANGING VECTOR AT $0308
110   ;
120   EXTBASON    LDA #<PRGSTART
130               STA <EXECVECT
140               LDA #>PRGSTART
150               STA >EXECVECT
160   ;
170               RTS
180   ;
190   ;
200   ;
210   ;TURN EXTENDED BASIC OFF
220   ;BY CHANGING VECTOR AT $0308
230   ;BACK TO NORMAL ($A7E4)
240   ;
250   EXTBASOFF   LDA #<$A7E4
260               STA <EXECVECT
270               LDA #>$A7E4
280               STA >EXECVECT
290   ;
300               RTS
310   ;
320   ;
330   ;
340   ;*** MAIN PROGRAM ENTRY ***
350   ;
360   ;EXECUTE A STATEMENT (LIKE $A7E4)
370   ;
380   PRGSTART    JSR CHARGET
390               JSR PRGSTRT1
400               JMP $A7AE
410   ;
420   ;LOOK FOR EXTENDED BASIC COMMANDS
430   ;
440   PRGSTRT1    CMP 'O
450               BEQ OFF.RT
460               CMP #$89      ;'GOTO'
470               BEQ GOTO.RT
480               CMP #$8D      ;'GOSUB'
490               BEQ GOSUB.RT
500   ;
510               CMP 'C
520               BNE NORMAL
530               JSR CHARGET
540               CMP 'O
550               BNE NORMAL
560               JSR CHARGET
570               CMP 'L
580               BNE NORMAL
590               JSR CHARGET
600               CMP #$B0      ;'OR' TOKEN
610               BEQ COLOR.RT
620   ;
630   ;
640   ;DO NORMAL ROM-ROUTINE
650   ;
660   NORMAL      JMP $A7ED
670   ;
680   ;
690   ;EXECUTE 'COLOR' COMMAND
700   ;
710   ;GET INK PARAMETER
720   ;
730   COLOR.RT    JSR CHARGET
740               JSR $AD8A
750               JSR $B7F7
760   ;
770   ;CHANGE INK COLOUR
780   ;
790               STY 646
800   ;
810   ;GET PAPER PARAMETER
820   ;
830               JSR CHARGET
840               JSR $AD8A
850               JSR $B7F7
860   ;
870   ;CHANGE PAPER COLOUR
880   ;
890               STY 53281
900   ;
910   ;GET BORDER PARAMETER
920   ;
930               JSR CHARGET
940               JSR $AD8A
950               JSR $B7F7
960   ;
970   ;CHANGE BORDER COLOUR
980   ;
990               STY 53280
1000  ;
1010  ;JUMP TO REST OF ROM-ROUTINE
1020  ;
1030              RTS
1040  ;
1050  ;
1060  ;EXECUTE 'GOTO' COMMAND
1070  ;
1080  GOTO.RT     JSR CHARGET
1090  GOTO.RT1    JSR $AD8A
1100              JSR $B7F7
1110              JSR $A8A3
1120              RTS
1130  ;
1140  ;
1150  ;EXECUTE 'GOSUB' COMMAND
1160  ;
1170  GOSUB.RT    JSR CHARGET
1180              LDA #$03
1190              JSR $A3FB
1200              LDA $7B
1210              PHA
1220              LDA $7A
1230              PHA
1240              LDA $3A
1250              PHA
1260              LDA $39
1270              PHA
1280              LDA #$8D
1290              PHA
1300              JSR GOTO.RT1
1310              JMP $A7AE
1320  ;
1330  ;
1340  ;TEST FOR REST OF 'OFF'
1350  ;
1360  OFF.RT      JSR CHARGET
1370              CMP 'F
1380              BNE NORMAL
1390              JSR CHARGET
1400              CMP 'F
1410              BNE NORMAL
1420  ;
1430  ;EXECUTE 'OFF' COMMAND
1440  ;
1450              JSR EXTBASOFF
1460  ;
1470  ;GET NEXT CHARACTER AND
1480  ;JUMP TO REST OF ROM-ROUTINE
1490  ;
1500              JSR CHARGET
1510              RTS
```

PRINT ENHANCER Plus/4 disk £19.99

Plus/4 desktop publishing: Proportionally spaced, high quality printouts, from BASIC, 3+1 and script/plus. Supplied with 15 fonts and font designer program. MPS 801/803, VIC 1525.

RS232 INTERFACE 64, 128 or Plus/4* . . . £34.99

Measuring only 12x69x57mm, the smallest and neatest unit available. Conforms to Commodore 1011A user-port standard. Compatible with BASIC, Easyscript, Superscript, Superbase, Mini Office, Script Plus etc. Connect to RS232 printers, modems and other devices. Supplied with 1 metre of cable (add 75p extra metre*) terminated with a male (female*) 25-way D-connector, or a custom cable made to your requirements. Our after sales service for unusual printers guarantees to get you up and running. Supplied with a terminal emulator program, file transfer and other utilities on **tape/disk***

IBM PC File Transfer Utility £9.99

Captures text or binary files via the COMI RS232 port. Supplied on a 360k IBM format 5¹/₄" disk. Simple instructions provided by on-line help.

VIEWDATA TERMINAL 64 or Plus/4 . . . . £14.99

Access Prestel, Micronet, CityService etc. Works with any user-port RS232 interface and 1200/75 modem.

CENTRONICS CABLE 64, 128 . . . . £18.99

Use with Easyscript, Superscript etc. Driver for BASIC on disk or tape.*

6502 ASSEMBLER 64 or Plus/4* tape/disk £12.99

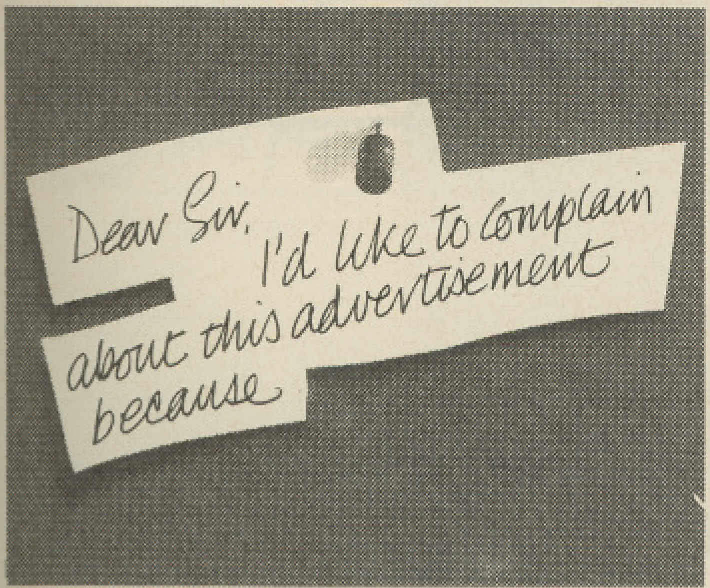
A sophisticated two-pass symbolic assembler and text editor which supports tape and disk filing. The assembler, editor, source and object code may all be resident in memory simutaneously, facilitating rapid and interactive code development. Very fast edit/Assembler/test cycles.

Z80 EMULATOR/ASSEMBLER 64 disk £12.99

A unique integrated Z80 development package. The emulator compiles Z80 source code into optimised 6502 which runs on the 64 at about one sixth the speed of a 2MHz Z80. The cross-assembler generates hex or binary Z80 object files. The disk contains a powerful editor and example programs.

Please specify* options. Send cheque/PO. or order by access. Prices include VAT and P&P. Overseas orders add £2.50. Allow up to 1 week for delivery.

YORK ELECTRONIC RESEARCH
The Fishergate Centre, Dept YC, 4 Fishergate, York YO1 4AB
Tel (0904) 610722



Most advertisements are legal, decent, honest and truthful. A few are not, and, like you, we want them stopped.

If you would like to know more about how to make complaints, please send for our booklet: 'The Do's and Don'ts of Complaining'. It's free.

The Advertising Standards Authority.
We're here to put it right.

ASA Ltd., Dept. Z, Brook House, Torrington Place, London WC1E 7HN.

This space is donated in the interests of high standards of advertising.

To find out all you need to

Know

about the Amiga computer

it

pays to subscribe to 'Your Amiga'

all

you need to do is fill in the coupon

U.K. £9.00, Europe £11.80, Middle East £11.90,
Far East £13.00, Rest of the World £12.10
Airmail rates on request.

Please commmence my subscription to YOUR AMIGA with the .... issue. I enclose my cheque/money order for £................ made payable to ARGUS SPECIALIST PUBLICATIONS LTD.
or debit £................. from my Access/Barclaycard

No. ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Valid from .... to ....

Signature ....

Name ....

Address ....

.... Postcode ....

Send this form with your remittance to:
INFONET LTD. 5 River Park Estate,
Billet Lane, Berkhamsted, Herts HP4 1HL

Basic Workshop

Avoid typing all your keywords out in full with this ingenious new program

by B. Graham

Basic Workshop is a program which lets you press two keys (CTRL + ?) to print a keyword such as 'LIST'. It allows the Commodore keyboard to be used in much the same way as the Spectrum keyboard. It is stored at $C000-C800, so as not to get in the way of Basic programs.

It works by using interrupts and the keyboard matrix: when the interrupts are serviced, the program checks to see if a key is pressed. If so, it stores the string in the keyboard buffer. The strings are held in a table, and most m/c language programmers should be able to change them with ease. The program can also change screen colours with the function keys, and has an inbuilt 'OLD' and Reset. This allows you to examine programs such as 'Patience' by Commodore by loading the program, resetting and OLDing the program.

It was written on the Monitor/Assembler on Action Replay IV. A full list of commands. etc, follows:

LOADING

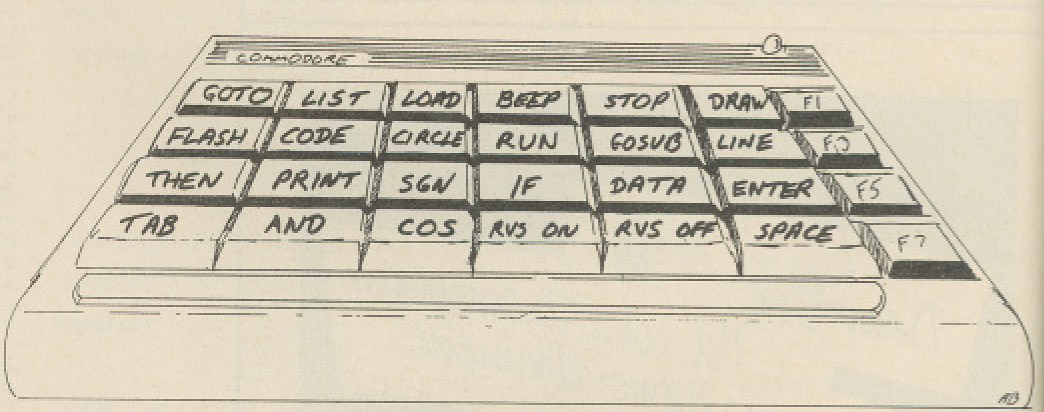
Type: LOAD"BASIC WORKSHOP",8,1
Then: NEW
Then: SYS49152
If this is not done when entering lines, all you will get are 'OUT OF MEMORY' errors. It should now print a message showing that all is well.

CONTROLS

All the controls can be seen by pressing CTRL + ↑ to show a helpsheet.

F1 = Cursor colour.
F3 = Screen text colour.
F5 = Screen background colour.
F7 = Border colour.

Shiftlock = Listing, directory, etc. Pause.



CTRL + A = ASC	CTRL + B = GOSUB
CTRL + C = CLOSE	CTRL + D = DIRECTORY
CTRL + E = PEEK	CTRL + G = GOTO
CTRL + H = CHR$(	CTRL + I = INPUT
CTRL + L = LIST	CTRL + O = OPEN
CTRL + P = POKE	CTRL + Q = LOAD
CTRL + R = RETURN	CTRL + S = SAVE
CTRL + T = DATA	CTRL + V = VERIFY
CTRL + + = OLD	CTRL + – = NEW
CTRL + ↑ = HELP	CTLR + (CBM) + = RESET

When OLDing you MUST press CTRL and +, and not type OLD. This is because it is the keypress which OLDs not the actual word. Thus you get a Syntax error if you type it in.

Games Update

Jocky Wilson's Darts Challenge

Being one of the favourite pastimes in the UK, a darts game just has to be part of any computer games lover's collection. There have been many versions made for the C64, this latest effort from Zeppelin games being endorsed by Jocky Wilson.

Numerous different games are on offer. 'Tournament' allows any number of players to compete in a real darts game. 'Two player head-to-head' allows just two players to team up against each other, and 'round the clock' is the old favourite where you have to hit the numbers in sequence. If playing round the clock, you can specify if you need to hit just the number, trebles or doubles in order to pass onto the next number in the sequence.

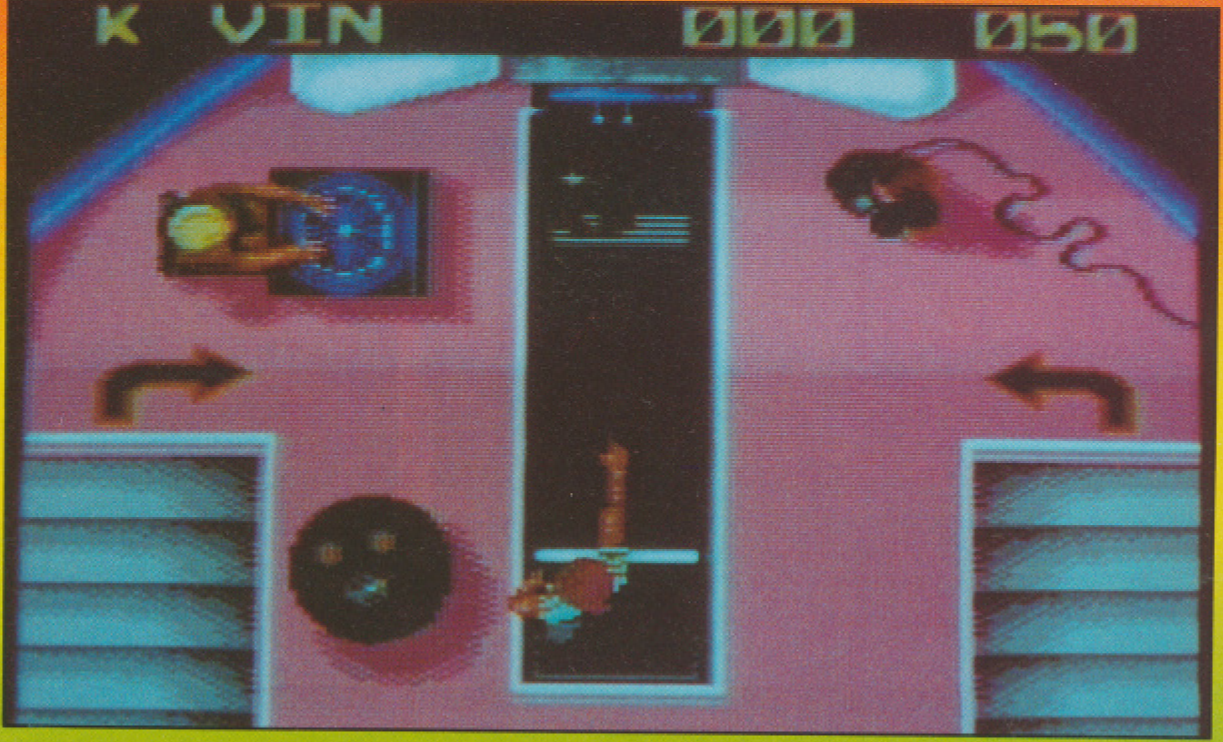
Once you've selected your playing options and are ready to throw, the screen changes to a display of a hand holding a dart in front of a dart board. The hand on display moves around in a semi-random pattern as though the dart holder is suffering from a bad case of the morning-afters.

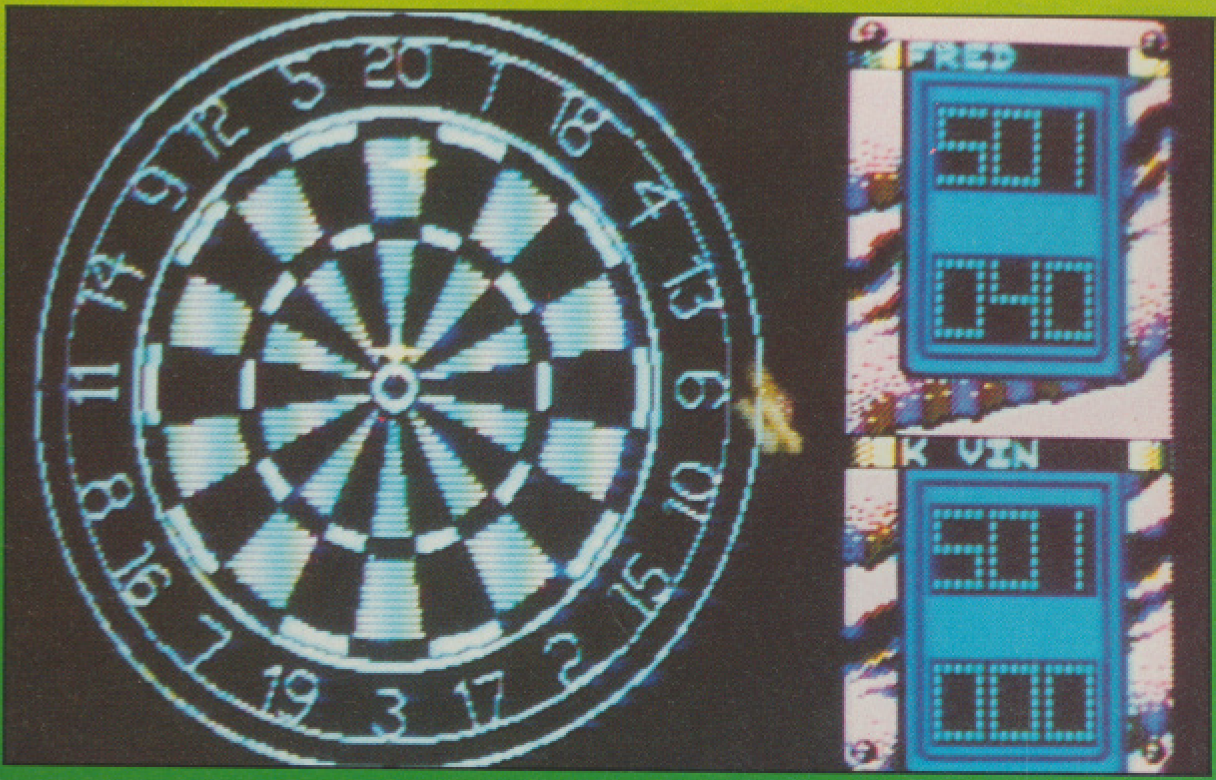
You control the rough direction that the hand moves in by the joystick. Simply press fire when the dart is over the position on the board that you'd like to hit, and sit back and watch your dart fly towards the board. That's all there is to it – you've no control of the strength of the throw or the angle at which the dart flies – both of which I would assume to be an essential part of any darts simulation.

If you're playing against the computer, the display will change to a 3D representation of the board. This is a purely cosmetic feature, and you simply sit back and watch the computer throw its three darts.

As darts simulations go, it's far from being the best that I have seen. All is not lost, though – the game is fun to play, even if it's not a true simulation. A budget price and the option to play against other people also help to lift this game from the "don't bother" league into the "it could be worth adding to your collection" one.

Touchline:
Name: *Jock Wilson's Darts Challenge.* ***Supplier:*** *Zeppelin Games, 28 Osbourne Road, Jesmond, Newcastle upon Tyne, NE2 2AJ.* ***Tel:*** *(091) 281 4401.* ***Price*** *£2.99.*





Jet Bike Simulator

Here's your chance to show off on a motorbike, and at a budget price, courtesy of Code Masters. *Jet Bike Simulator* allows you to compete in jet bike races on one of 24 courses. Up to two human players can compete against the computer-controlled bikes, but with keyboard only – no joystick option is given.

Though the action in this game is fast and furious, it is spoilt a little by the screen display. In these days of scrolling screens, split screens and 3D views, Code Masters seem to have taken a backward step with this game, as each circuit fits onto the monitor screen. This doesn't really spoil the play – if anything it makes it easier, as you can see what's coming up – but it does mean that all the jet bikes appear on the screen as slightly oval dots, making differentiation between them difficult.

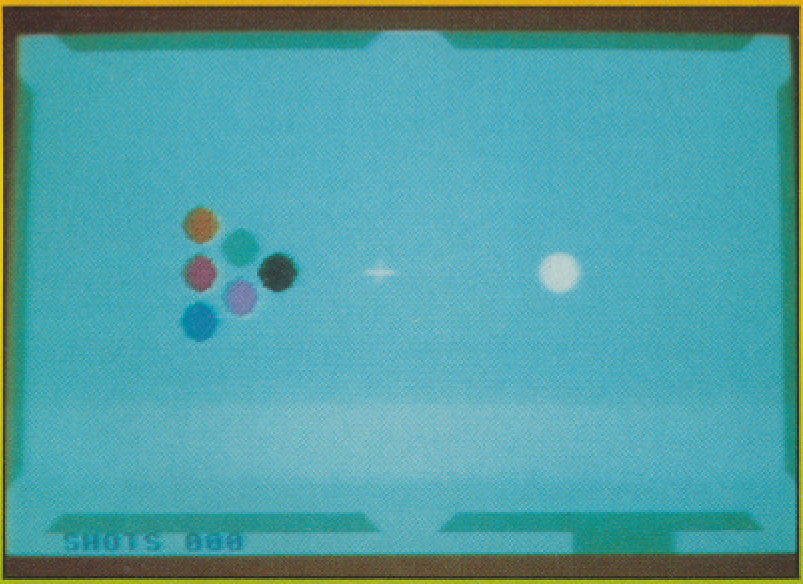
I must admit to never having had a go on a jet bike. However, if the control of a real jet bike is as difficult as it is with these computerised ones, I don't really fancy giving it a go. Some may think that travelling along at full speed and turning right to go through the next gate will be easy – if so, you'll be in for a shock. When turning a corner, you can't forget about the forward momentum of your bike. If you do, you're liable to find yourself embedded in the nearest bank.

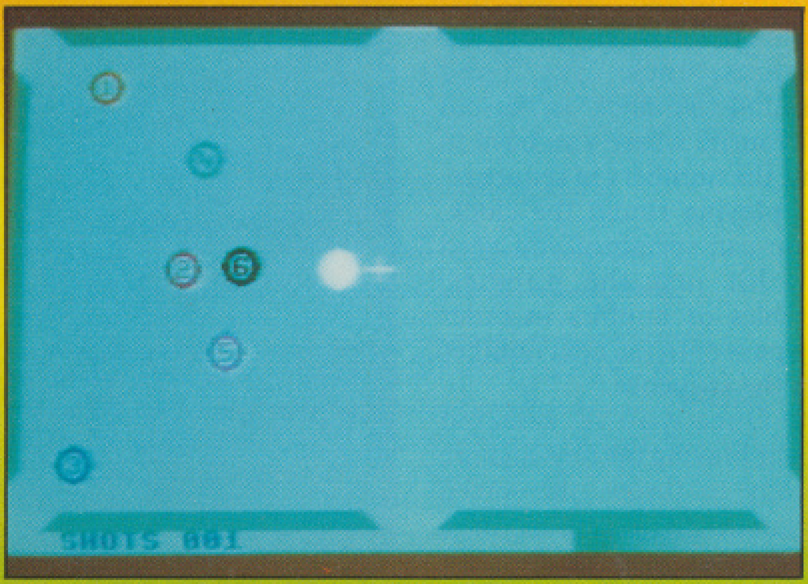
Once you've mastered the normal version of *Jet Bike Simulator*, which if my performance is anything to go by will take quite some time, Code Masters have provided you with an advanced version. This uses the same courses, but offers extra challenges.

Somehow, *Jet Bike Simulator* doesn't quite make it to the status of being one of those "just one more go" types of games. It's good fun, the two player version especially so. Perhaps it's the poor display that causes the game to lack that final polish.

Touchline:
Title: *Jet Bike Simulator.* ***Supplier:*** *Code Masters, PO Box 6, Leamington Spa, CV33 OSH.* ***Tel:*** *0926-814121.*

Hustler





There are some games that the computer is just right for, while others are utterly destroyed by being transferred onto the electronic wizard. *Hustler,* subtitled *World Championship 6 Ball Pool,* seems to be one of those conversions that doesn't live up to its real-life counterpart.

Any pool or snooker player will tell you that the skill in playing the game depends upon the ability of the player to position the balls just where he wants them. This is done not only by hitting the cue ball at varying strengths, but also depends upon exactly where you hit the cue ball with the cue.

Hustler, from Top Ten, allows you to select where you would like to send the cue ball and at what strength, but that's all. In fact, selection of the strength of the shot isn't so much a choice as a test of your reactions. To select the strength of the shot, you're required to watch a power bar at the bottom of the screen and press the fire button when it's at the position you require.

Hustler gives the player a choice of six different pool games: one player – any ball in any pocket; one player – pocket the balls in order; one player – put each ball in its correct pocket; two players – put each ball in its pocket and score the pocket; two players – mini pool, one player to go for the orange balls, the other black; two players – one pots in order 1-6, the other pots in order 6-1.

This simulation offers a good selection of different games, and the two player games are fun when competing against a friend. Unfortunately, the game lacks the realism that would make it a top-notch pool game. As it stands, I'd rather put my money into the real tables where there's a much greater skill element involved, though the £1.99 price tag may just make it attractive enough to add to your collection for those damp, cold nights when you can't be bothered to go out.

Touchline:
Name: *Hustler.* ***Supplier:*** *Top Ten, MIL, 12 Chiltern Enterprise Centre, Station Road, Theale, Berks. RG7 4AA.* ***Tel:*** *0734-302600.* ***Price:*** *£1.99.*

Super Stunt Man



If your idea of fun is driving a car through a bolder-strewn desert at high speeds, racing a powerboat through tight twists and turns, getting back into your car in order to drive as fast as possible through a forest, turbo boosting over the grand canyon, fighting it out with violent street gangs, and so on, then Code Masters have just the job for you as a stuntman in their latest game.

Your job description as a stunt man in Code Masters' latest epic will be to navigate your way through all of the challenges mentioned above within a certain time limit. You have just four takes at each scene, and you must complete each one in turn. Of course if you fail in your task, then you'll have to start all over again, as the film is no good.

Super Stunt Man is one of those fast action games that makes you feel sure that just one more go will give you a better score. Or perhaps you feel sure that you've mastered the speed boat racing, and one more go will see you through it and onto the next stage of the game.

If you're the sort of person who can't stand finding out that the time is 3 am and you're still playing that stupid game, but you'll have just one more go, then don't buy this program. On the other hand, it's a must for all addicts of fast action arcade games.

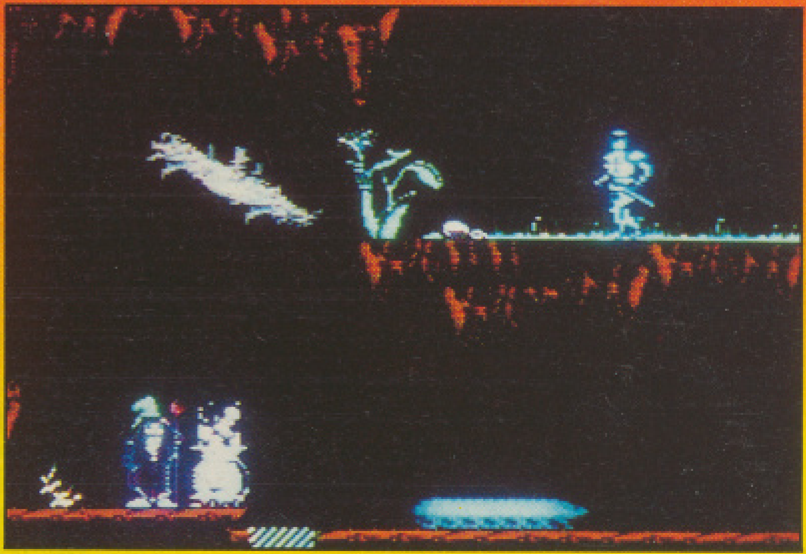
Touchline:
Title: *Super Stunt Man.* ***Supplier:*** *Code Masters, PO Box 6, Leamington Spa, England, CV33 0SH.* ***Tel:*** *0926-814121.*

Terra Fighter

Time to save a defenceless Earth from a horde of marauding sprites again, and at a budget price. *Terra Fighter* is a sideways scrolling shoot-em-up containing six levels. However each level has 18 distinct stages, so the game will take some getting through. The idea is to clobber the baddies' power supply, while dodging the hazards. We particularly liked the one that reverses your joystick controls.

Touchline:
Title: *Terra Fighter.* ***Supplier:*** *Zeppelin Games, 28 Osborne Road, Jesmond, Newcastle upon Tyne, NE2 2AJ.* ***Tel:*** *(091) 281 4401.* ***Price:*** *£1.99 (cass).*

Camelot Warriors



Time to get the old broadsword honed and ready. *Camelot Warriors* is a graphic adventure of some complexity, but there's still room for the hack and slayers. If you can make it through, its a matter of getting the four Magic Things to the Four Magic Kings. Not easy, and reasonable for a budget game.

Touchline:
Title: *Camelot Warriors* ***Supplier:*** *Mastertronic, 2-4 Vernon Yard, Portobello Road, London W11 2DX.* ***Price:*** *£1.99 (cass).*

How to be a Complete Bastard

After all, it's a useful skill. Most companies have many openings for CBs, largely in management. Ha, ha – only kidding.

This is, of course, the budget version of last year's Adrian Edmondson gross-out. This game is almost the return of Bad Taste software. All you have to do to clear most of the yuppies out of a rather dull party by being totally gross, something at which we journalists excel. Stealing, setting fire to things, frightening people, and a range of far less mentionable activites, are all possible. Format is fairly standard graphic adventure, but this game has enough novelty to lift it above the crowd.

Touchline:
Title: *How to be a Complete Bastard.* ***Supplier:*** *Mastertronic, 2-4 Vernon Yard, Portobello Road, London W11 2DX.* ***Price:*** *£1.99 (cass).*



Pilot - a Programmer's Language

If you're interested in educational programming but have yet to find a suitable language, Pilot could be the answer

By Evelyn Mills



Pilot is a language designed for those who are interested in educational programming. This particular version (Commodore 64 Basic V.2) is extremely versatile, easy to use and has approximately 38,900 bytes available for programming.

The concept of Pilot is frequently a simple 'question and answer' program, which in reality is not the case. Quite apart from all the important dialogue elements there also exists:

- a simplified graphics screen with full colour control;
- excellent sprite designing and handling procedures;
- the ability to change all keyboard characters to any design you wish;
- sound programming availability.

First of all, the simplicity of 'dialogue' programming should be studied. There are three modes of operation – EDIT, COMMAND and IMMEDIATE. (The latter is essentially for 'practising while learning'.)

In EDIT mode (press E key), text may be entered directly in the form of a wordprocessor; printer output will be limited, however, to the normal 40 column screen width. More importantly, in EDIT mode, is the facility to delete text, enter text and generally make amendments as required.

Real Time Programming

To get going with 'real time' programming, begin in COMMAND mode (press STOP key) and use this in conjunction with the EDIT mode; (all the PILOT modules are interactive). Consider a simply example such as:

```
TS:  What is your name please?
A:
T:   Hello, $%B. press RETURN to
     continue.
A:
```

TS starts the typing instruction sequence which is subsequently followed by a T; the words typed in will appear on screen. A accepts the words or instruction typed by the user. New lines are simply preceeded by a colon which is a legal continuation instruction.

Naturally there must be flexibility of control while programming otherwise the dialogue will be nothing more than monotonous. PILOT provides ample scope for this and has restructured the more advanced features of Basic into a simple format. A sample of such features is as follows; each instruction must be followed by a colon (:) not listed here.

PR – problem solving
D – dimension
W – wait
M – match
J – jump instruction telling the computer to jump a label inserted within the program
C – string computation

To avoid answers which cannot be anticipated, hints may be inserted to combine with the match, type and jump commands. U (use) calls subroutines which are concluded with an E.

When Numerical Answers are Required

Questions which require numerical answers are streamlined to a high degree, e.g. X=RND(90) returns a number between 0 and 89. All the calculations handled by the C64 are permissible and are computed using an AND operator which automatically checks for the correct answer.

The manual is full of examples which will take you through every aspect of Pilot programming speedily and efficiently. You have here a a very powerful tool for education (and fun!!) working with simple structured commands.

Graphics

Turning to the GRAPHICS side we start by entering G:E followed by the requisite field instructions. The *x* and *y* axes are 319 by 192 respectively, the origin (0,0) being in the lower left hand corner. Initially the starting point should be defined (P) followed by the 'draw to' locations (D). To draw a triangle, the command could be formatted as:

```
G:E;P5,5:D90,5;D90,70;D5,5
```

The graphics beam may be moved to another point location using an M, while lines may be erased with an R command; both of these commands need, of course, the requisite *x,y,* locations.

In combination with this, the text cursor may be positioned exactly where required, the screen may be split using an S command and linked to Frame or Window Designing for maximum display effect. The full colour range of the C64 is available with single letter instructions for border (X), background (B) and foreground (F) colours. The FILL command (F) is used to fill a block with a foreground colour by specifying the diagonal co-ordinates in the normal manner (e.g. G:E:P50,50;F160;C14)

Pilot again comes into its own by allowing you to design sprites on screen and to move them around with simple instructions. The BIT-PATTERN (B) is used to define your sprite on a 21 x 24 grid, using a dot to create the grid and an *x* to indicate the pattern. The sprite should be given a number (0-7). To start the program type B:2 and go ahead with your pattern (2=sprite number !!). Having done so, the sprite must now be turned on with an S command and an op code (E1). An example is as follows:

```
S:2:E1;C0;L100,100
W:100
```

The sprite will now be apparent, coloured black, at location 100,100 on your screen.

Sprites may be expanded horizontally or vertically, restored to normal size, or used in multi-colour mode (M1) with three possible colours in any one sprite. Again full instructions are given in the manual combined with the simple commands for moving sprites around the screen.

Character Definition

There are many applications in which specialised notation would be useful, particularly in foreign languages, maths, science, etc. Using Pilot a new character may be defined by using the NEWCHAR (N) instruction; the N is followed by a number corresponding to the standard ASCII code of the keyboard (32-127). The code for $ is 36 for example, so N:36 will allow redefinition of this key. Redefinition is done on an 8 x 8 matrix in much the same manner as sprites.

Sound

This enhancement is very useful for creating extra effects in dialogue or animation, etc, but can only be regarded as a supplement and will not give you any degree of musical flexibility. Sirens, bells, locomotives and so forth may be programmed with ease to good effect and, again, the method is simple enough. Noise, pulse, triangle and sawtooth waveforms are available, as are ring modulation and synchronisation. Volume and duration of notes, attack, decay, sustain, release are all there and a sound editor is available on the disk to allow you to experiment with the above variables. Music note values are listed in the appendix.

There are more advanced features in Pilot which show you how to set up multiple choice questions with selected answers (look under EXECUTE), while the ESCAPE mode allows special features to be built up within a program and called by the user at any time. A good example is given of how to create a calculator which may be called at any point within a program.

In the event that you wish to create a long program and run out of memory, there is a LINK instruction (L:PART 2) which will load the second part, (and subsequent parts if specified), automatically. Naturally the previous program is then unavailable so the sub-sections should be independent as such.

The normal features of saving, loading, printer output are all available; a "Run Only" version of Pilot is on the disk which means that your program may not be listed, edited or printed. The disk has an excellent demo, which when listed, will show you how to handle sprites, graphics, split screens and programmable characters.

For the programmer, Pilot is an easy to use, interactive system which can be well recommended; nor will this cost you a fortune; contact Dimension Computers Ltd.

Tank Attack





With wargames becoming ever more sophisticated, it's good to see that one company – CDS – has not forgotten the beginner. What is doubly unusual is that their latest title, *Tank Attack*, is also a curious hybrid: a computer moderated wargame.

The scenario is simple. Four neighbouring countries, Armania, Kazaldis, Sarapan and Calderon, declare war on each other – well, on two of the others actually, for there are two firm alliances. Resources are limited to armoured vehicles, and the object is to blow holes in your enemy's command post before he does likewise to you. The game can be played by two to four players. There is no solo option.

The battle itself is fought out on a board by moving little plastic tanks round a hex grid. Each country has two light and medium armoured cars, as well as two light, medium and heavy tanks. The pieces are cunningly designed so that you can't tell the strength of an opposing piece when it's on the board – you can only find out by attacking it. The pieces can be placed anywhere you want within the boundaries of your country. It's up to you to come up with the right balance of attack and defence.

How far you move each turn is determined by the computer taking into account the weather for the day, and assorted other random factors. The number of hexes that you can move can be split between as many vehicles as you see fit. There are a few terrain restrictions and a bonus for armoured cars in open country.

If you want to attack, you must inform the computer how far away from your target you are, and then disclose what power vehicle you have. Your enemy does likewise, and the computer then works out the result of the battle. At this stage, you get to see the battle through a stylised pair of binoculars on the screen. (Incidentally, I wonder if the game designers have ever actually looked through binoculars. The sideways figure of eight view is not what you get. Hollywood directors please take note too.) The result of the battle is then displayed, effects ranging from total destruction to a vehicle being removed to a repair centre.

That's really all there is to the game. The board looks attractive, and the on-screen commands are clear and simple to use, but (and it's a huge but) why bother in the first place? The game would have worked just as well without the computer being there. All you need to do is determine the move factor randomly, and look up the results of combat on a table.

If, however, the game had been entirely controlled by computer, it could have been truly excellent. All the problems associated with board wargames could have been eliminated at a stroke, and a host of other features included – a save game facility should something interrupt the battle; a solo option; hidden movement; deliberate smoke screen effects; and, dare I say it, the elimination of cheating, or at least the urge to cheat. The idea of getting the computer to tell you how far you can move, and then making the player go and look up in the rule book exactly where he can move, does seem to be something of a retrograde step.

Tank Attack is extremely well produced, but I feel that it is the wrong game for this combination of board and computer action.

Touchline:
Title: *Tank Attack* ***Supplier:*** *CDS, CDS House, Beckett Rd, Doncaster DN2 4AD.* ***Price:*** *£12.99 (cass), £14.99 (disk)*

"THE MAIN EVENT"

DEPT (YC) 61 STAFFORD STREET, HANLEY, STOKE-ON-TRENT, STAFFORDSHIRE ST1 1LW

24 HOUR ANSWER SERVICE. TEL: 0782 281544

COMMODORE 64	
OLLIES FOLIES	1.75
WEB	£1.75
REAL YOU	£1.75
SLURPY	£1.75
AMAVROTE	£1.75
ARCADIA	1.75
GALATIC GARDENER	1.75
WORLD CUP FOOTBALL	1.75
PYRAMID	1.75
BEAKY & EGG SNATCHER	1.75
GOLF	1.75
TITANIC	1.75
RATSPORT	1.75
HELION	1.75
AM CONDOR	1.75
EGBERT	1.75
FABULOUS WANDA	1.75
WILD WEST	1.75
EUREKA	1.75
SPYS DEMISE	1.75
STREET MACHINE	1.75
ZIP	1.75
DESTRUCT	1.75
OSMIUM	1.75
AQUANAUT	1.75
DANTES INFERNO	1.75
HARTLAND	1.75
PAZZAR	1.75
PCW GAMES COLL	1.75
BIOLOGY	1.75
GEOGRAPHY	1.75
HISTORY	1.75
GERMAN	1.75
GORTEX	1.75
QUAKE	1.75
PSI WARRIOR	1.75
ANRIGA	1.75
VELOCIPEDE	1.75
FUNGUS	1.75
DESERT HAWK	1.75
RONALD ROBBERDUCK	1.75
ELECTRIX	1.75
F.A. CUP	1.75
TOMAHAWK	1.75
BATALYX	1.75
SCROLLS OF ABADON	1.75
CRAZY COASTER	1.75
TOAD FORCE	1.75
CLEAN UP TIME	1.75
F.A. CUP 87	1.75
FORCASTER	1.75
PROJECT PLANNER	1.75
NUMBERS AT WORK	1.75
DECISION MAKER	1.75
ENTREPRENEUR	1.75
FLYING FEATHERS	1.75
KICK OFF	1.75
CO LEVEL	1.75
CODE NAME MARK 2	1.75
CHINESE JUGGLER	1.75
ENCOUNTER	1.75
NIGHT SHADE	1.75
BROAD STREET	1.75
PYRAMID/TITANIC	1.75
BEAKY/RATSPORT	1.75
SOFTWARE STAR	1.75
ROCK N BOLT	1.75
MORDENS QUEST	1.75
FORBIDDEN FOREST	1.75
PANIC	1.75
ZIM SALA BIM	1.75
HAMPSTEAD	1.75
SENTINEL	1.75
ADRIAN MOLE	1.75
ENIGMA FORCE	1.75
STARION	1.75
ASSAULT MACHINE	1.75
WARRIOR II	1.75
STOCK CAR	1.75
CYBERTRON MISSION	1.75
BUMBLEBEE	1.75
FELIX IN FACTORY	1.75
FRENZY	1.75
GHOULS	1.75
SWOOP	1.75
ELECTION	1.75
SHERLOCK HOLMES	1.75
PARADROID	1.75
HIDE AND SEEK	1.75
MOON SUTTLE	1.75

COMMODORE 64	
SUPERCAN	2.50
B 24	2.50
TOOTH INVADERS	2.50
MISSION 2	2.50
HENRYS HOUSE V 1	2.50
HENRYS HOUSE V 2	2.50
TURBO 64	2.50
PILOT 64	2.50
SUPERMAN	2.50
DANGERMOUSE	2.50
JET SET WILLIE	2.50
NEXUS	2.50
EVIL CROWN	2.50
CHUCKIE EGG	2.50
MANIC MINDER	2.50
BOUNTY BAR	2.50
ON COURT BASEBALL	2.50
BLACK WYCHE	2.50
SKY RUNNER	2.50
BOULDERDASH	2.50
BOULDERDASH 2	2.50
AVENGER	2.50
FUTURE KNIGHT	2.50
RANA RAMA	2.50
HYPABALL	2.50
DANDY	2.50
ELEVATOR ACTION	2.50
NUCLEAR EMBARGO	2.50
MISSION AD	2.50
INT KARATE	2.50
TRAP	2.50
PRODIGY	2.50
EXPLORA	2.50
PUB GAMES	2.50
GREYFELL	2.50
DEACTIVATORS	2.50
GOBOTS	2.50
HYBRID	2.50
MOUNTIE MICK	2.50
TRIAXAS	2.50
PILE UP	2.50
WERNER	2.50
CHAMELEON	2.50
EIDOLON	2.50
CAPTURED	2.50
GOLDEN OLDIES	2.50
ALIENS	2.50
HOWARD THE DUCK	2.50
TRAIL BLAZER	2.50
FIFTH SQUADRON	2.50
SNAP DRAGON	2.50
HUNCH BACK	2.50
KRAKOUT	2.50
SCUBA DIVE	2.50
WORLD SERIES BASEBALL	2.50
KONG STRIKES BACK	2.50
SYSTEM 1500	2.50
RAID	2.50
THE FORCE	2.50
MANDRA GORE	2.50
MIND SHADOW	2.50
LITTLE COMPUTER PEOPLE	2.50
IMHOTEP	2.50
MURDER ZINDER	2.50
YOUNG ONES	2.50
TOUCHDOWN FOOTBALL	2.50
STARSHIP	2.50
GYROSCOPE	2.50
LEVIATHAN	2.50
BALLBLAZER	2.50
DEUX EX MACHINA	2.50
MERMAID MADNESS	2.50
FRACTALUS	2.50
MAGIK	2.50
ADDICTABALL	2.50
LIGHTFORCE	2.50
MEGA COMPILATION	2.50
RADAR RAT RACE	2.50
SPINDIZZY	2.50
TWO ON TWO	2.50
BASEBALL	2.50
STAR RAIDER 2	2.50
FIRE TRACK	2.50
ACTION PACK II	2.50
ULTIMATA	2.50
MAGIK MADNESS	2.50
JAIL BREAK	2.50
BOTHAMS T. MATCH	2.50
RFV WIED MONTY	2.50
CONVOY RAIDER	2.50

COMMODORE 64	
WINTER GAMES	6.99
FOOTBALL MANAGER 2	5.45
R-TYPE	6.90
NINJA 2	8.99
BUBBLE BOBBLE	6.95
SALAMANDER	6.25
WE ARE CHAMPIONS	6.99
D.T. PLYMPIC CHALLENGE	6.99
SUMMER OLYMPIAD	6.99
PSYCHO PIG OXB	6.99
REX	6.29
FERNANDUZ MUST DIE	6.99
G.L. HOTSHOT	6.99
G.L. SUPERSKILLS	6.99
SUPER HANG-ON	6.99
ULTIMATE GOLF	6.99
ROY OF THE ROVERS	6.99
SUPERSPORTS	6.99
MAD MIX GAME	5.59
LANCELOT	10.50
GREAT GIANA SISTERS	6.99
SUMMER OLYMPIAD	6.99
OVERLANDER	6.99
CALIFORNIA GAMES	6.99
CHAMPION SPRINT	6.99
DRAGON NINJA	6.29
UNTOUCHABLES	6.29
OPERATION WOLF	6.29
CYBERNOID II	6.99
INTENSITY6.95	
ECHELON	8.99
TYPHOON	6.29
GAME OVER 2	6.29
BUBBLE GHOST	6.99
BUTCHER HILL	6.99
ARTURA	6.99
TECHNO COP	6.99
MOTOR MASSACRE	6.99
GAPLUS	6.29
VINDICATOR	6.29
ARMALYTE	6.99
TARGET RENEGADE	6.29
L.A. CRACKDOWN DISK	10.50
THE BOBBY YAZZ SHOW	6.29
CHUBBY GRISTLE	6.95
WONDERBOY	6.95
POWER PYRAMIDS	6.95
VINDICATOR	6.29
VICTORY ROAD	6.29
STRIP POKER II	5.59
MANIAX	5.59
HAWK EYE	6.99
PINK PANTHER	6.99
SKATE CRAZY	6.99
BLOOD BROTHERS	6.99
G.I. HERO	6.95
BIONIC COMMANDO	6.99
ROAD BLASTER	6.99
DREAM WARRIOR	6.99
MICKEY MOUSE	6.99
ALIEN SYNDROME	6.99
DESOLATOR	6.99
PLATOON	6.99
E EDWARDS SUPERSKI	6.99
STREET FIGHTER	6.99
SPECIAL FX	6.29
BARBARIAN II	6.99
ARKANOID II	6.99
GAUNTLET II	6.99
720	6.99
CHERNOBEL SYNDROME	6.99
OUTRUN	6.99
HUNTED MOON	6.95
FIREFLY	6.29
BEYOND ICE PALACE	6.99
GRYSOR	6.29
MARAUDER	6.99
EVERY SECOND COUNTS	5.59
WIZARD WARS	6.99
COMBAT SCHOOL	6.29
IKARI WARRIORS	6.95
DRAGON'S LAIR	6.95
CHARLIE CHAPLIN	6.99
ALT WORLD GAMES	6.99
SHACKLED	6.99
TYPHOON	6.99
BARDS TALE	6.99
ACTION FORCE II	6.99
T-WRECKS	6.99
GUTZ	6.29

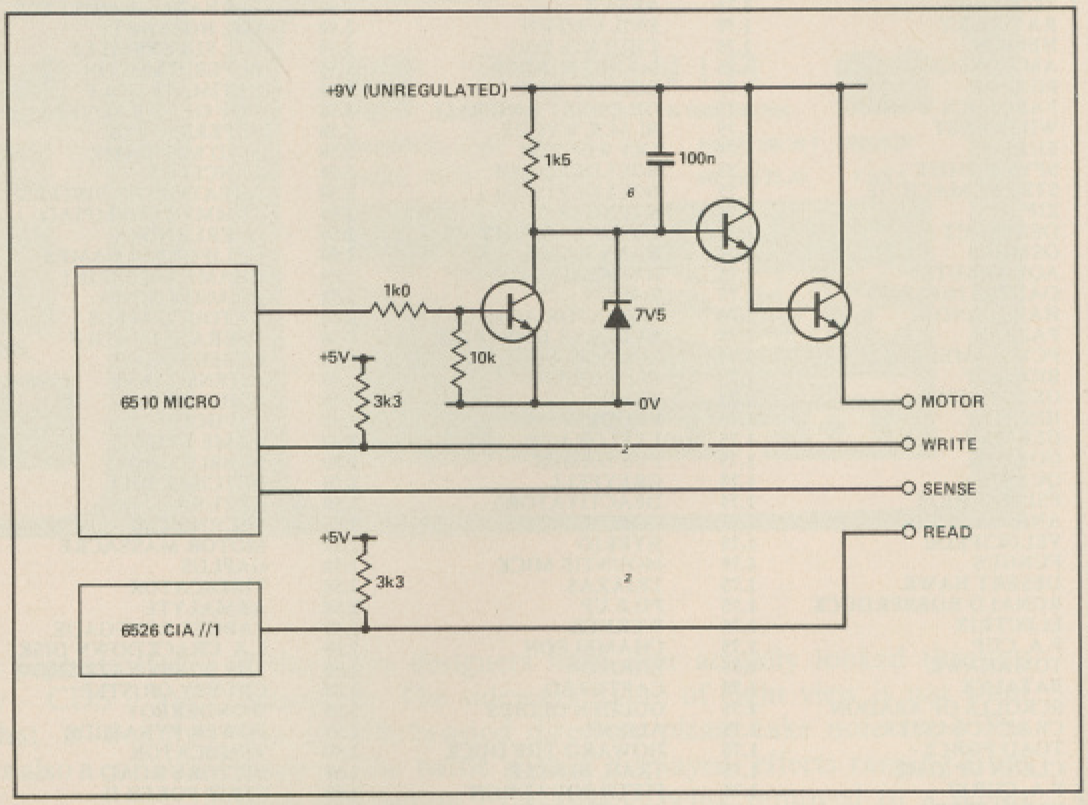
COMMODORE 64 DISK	
ALTER EGO FEMALE	7.95
ALTERNATE REALITY	6.90
BAKER STREET	4.95
GUNSLINGER	4.50
CROSS CHEQUE	4.50
OCTAPUS	4.50
G.F.L. FOOTBALL	5.95
CAPTAIN AMERICA	4.50
KNIGHT GAMES 2	4.50
EXPRESS RAIDER	3.95
DEATHSCAPE	4.95
FUTURE FINANCE	4.95
PROJECT PLANNER	1.95
FORECASTER	1.95
NUMBERS AT WORK	2.95
REAL YOU	1.95
DECISION MAKER	2.95
FANTASY FIVE	2.95
DANDY	3.95
SUSPENDED	2.95
WILD WEST	3.95
BUG BUTZ	3.95
MURDER ZINDERNOF	3.95
TOUCHDOWN FOOTBALL	3.95
BASKETBALL	3.95
GOBOTS	4.95
HYBRID	4.95
PILE UP	4.95
BLACK MAGIK	4.95
WINTER GAMES	4.95
ELITE COLLECTION	5.90
INT KARATE	4.95
HEAD COACH	4.95
PRESIDENT	4.95
GOLDEN OLDIES	4.95
PUB GAMES	4.95
FOOTBALL MANAGER	4.15
ARAC	3.95
MANDROID	4.50
PLASMANTRON	
OINK	4.50
TRAXXON	4.50
THUNDERCROSS	4.50
JACK THE NIPPER	4.50
DEATHSCAPE	4.95
FRANKENSTEIN	4.50
TIME FIGHTER	4.50
NINJA MASTER	4.50
DISCOVERY	4.50
JET BOYS	4.50
WOLFMAN	4.50
LIFE FORCE	4.50
I ALIEN	4.50
CUT & PASTE	8.95
CHALLENGE/GOBOTS	4.95
ENDURO RACER	10.50
500cc GRAND PRIX	10.50
LUCAS FILM COMP	10.50
PREDATOR	10.50
SPORTS COMPILATION	21.00
INDOOR SPORTS	10.50
WONDERBOY	10.50
THEY STOLE A KILLION	10.50
5 STAR GAMES 2	10.50
5 STAR GAMES 7	10.50
LIVING DAYLIGHTS	10.50
SKATE CRAZY	10.50
SHACKLED	10.50
MIND FIGHTER	13.99
MARAUDER	10.50
HOPPIN MAD	8.99
DREAM WARRIOR	10.50
DARKSIDE	8.99
DARK CASTLE	8.99
BLOOD BROS	10.50
BLACK JACK ACADAMY	17.50
BIONIC COMMANDOS	8.99
NEWSROOM	29.99
GAMES MAKER	13.99
PRINT MASTER	21.00
TASWORD	13.99
PRINTSHOP	29.99
TOY SHOP	39.99
VIDOE TITLE SHOP	10.50
MUSIC STUDIO	14.99
STREET FIGHTER	10.50
ROAD BLASTER	10.50
STANDING STONES	10.50
MERCENARY	6.95
I.D	8.99
ELITE	13.50

Please cross PO/Cheques and make payable to: THE MAIN EVENT.
To help maintain our low prices we would be obliged if you could include a first class stamp with your order, although this is NOT compulsory. Free Post & Packaging. Overseas please add 75p per tape.
We apologise for any alteration or omissions since going to press. The Main Event is a division of Software City.

Tech Troubles

Don't get stuck in a rut. Let our agony uncle help you out

By Tim Arnot



Commodore cassette schematic

Dear Tim,
I am trying to get my Commodore 64 to sample sounds from the cassette drive but have encountered a stone wall in actually getting it to read the signal coming in. I would like to know how to do this, and would be very grateful if you would send me some information on this subject or put an article in your magazine. I enclose a stamped addressed envelope and I look forward to your reply.
N.D. Moose, Chislehurst.

Hi Neil,
Thanks for including an S.A.E. As a general point, after your letters reach the *Your Commodore* offices, it can be up to a month before they reach me for reply, it will then be another month before they appear in print. So if you want a reply 'quickly' I will always send a copy of my reply in an S.A.E.

Unfortunately (for you anyway) the Commodore cassette system is entirely digital and relies purely on the time spacings of pulse transitions for receiving its information. While this makes it extremely reliable for its job of loading and saving programs, for your application of digitising sounds, it is utterly useless.

The Commodore cassette mechanism basically records information as a series of digital pulses of different length. A mixture of these lengths is then used to determine whether a bit is a 0 or 1. Thus a 1 is represented by a short pulse followed by a medium pulse, a 0 is medium followed by a short, and the start of a new byte is long followed by medium. I have included a brief schematic of the cassette read/write logic, so you can see what happens. Notice that the READ signal is fed directly into the FLAG input of CIA #1 ($DC00-$DCFF), which is basically just an edge sensitive interrupt input.

In order to sample sounds, you will need to employ a device called an Analogue to Digital Converter (ADC). This basically takes a snapshot of the sound at one instant in time and converts it into a number for use by the computer. The quality of the sampled sound is dependent on the speed at which you take these snapshots. For example, compact disk, which is essentially sampled sound, has a sample rate of 28,000 samples per second.

To show you how to construct a sampler based on an ADC is really beyond the scope of this reply, however you can purchase them ready made from a number of vendors, most of whom advertise regularly in this magazine. Cost would be in the area of £50, but that will also include software to drive it.

Dear Tim,
I own a Commodore Plus/4 system, and have a question which I hope you may be able to answer: how do I go about programming the user port? – either from BASIC or assembly language?

I am fairly conversant with the method used for the Commodore 64, so

if it is a similar method, all I need is the memory locations and any other differences.
Steve Jones, Gillingham

Hi Steve,
Unfortunately, I have bad news for you – the Plus/4 doesn't have a user port! – at least not in the sense that the PET, 64 and 128 have user ports.

To explain, the 64 etc, machines have an edge connector at the rear labelled Parallel user port, which provides a general bi-directional I/O port that can be used for general interfacing applications. And indeed we see a range of peripherals that use this port, from teletext adaptors to Centronics printer drivers. However, Commodore really intended to implement this port as an RS232 interface, using a 6551 Asynchronous Comms chip. Due to an ordering snarl up, they couldn't get the ICs, and so the parallel port stayed and RS232 was implemented in software.

In the UK, if we want RS232, we normally buy an interface that plugs into the cartridge port or serial bus, and contains said 6551 and some decent drivers. This is because the built-in port can't handle more than about 300 baud reliably. But in the States, all their modems and things plug into the built-in RS232.

And so to the Plus/4. At this time, Commodore finally got their act together and produced a real RS232 port using the proper hardware, running at decent speeds (up to 19,200 baud). This is what you have on the edge connector at the back of the machine. It is the same pinout as the 64 (from the RS232 viewpoint) so that the 64 async peripherals will plug directly in. This might seem to be of minority interest to UK users who want the parallel port, but unfortunately 75% of the users are Americans who want RS232.

Dear Tim,
I have typed in several games on my C64 which are for a friend's son and are for a Plus/4.

I appreciate that they will not run on the 64 and that certain items like flash commands must be added after they are loaded into the Plus/4 but how do I get them to load and list? I obviously need to change the start point for BASIC but how do I do this, and what should it be?
D.J. King, Huntingdon

Hi Mr King,
From the BASIC point of view, the essential differences between a C64 and a Plus/4 is the amount of memory available, the additional sound, graphics and disk keywords of the Plus/4, and the start location for BASIC text.

Of these three, the memory aspect can be discounted, since you are writing on the machine with less memory to transfer onto the machine with more.

The start of BASIC, similarly is not really important, as all BASIC programs on the C64 and Plus/4 (and C16 and 128 machines) are designed to relocate themselves to wherever the start of BASIC might be. The upshot of this is that LOAD "PROG-NAME",8 will work on any machine except the PET. As to where the start of BASIC sits, on the C64 it is at location $0801, and on the Plus/4 it can be in one of two places, depending on whether you have a graphics screen active or not. These two locations are $1000 (without graphics screen) and $2000 (with graphics screen).

When you type a line of BASIC and press the return key, the computer scans what you have typed for words that it recognises and converts them into one or two byte tokens. This was originally purely to save memory, since a token takes up much less space than the word itself. Any words that it does not recognise are assumed to be variables and so are left alone.

When you run the program, these tokens are indexed into a lookup table which points to the section of ROM that performs the function you are asking the computer to do. If the keyword has not been tokenised (for instance typing FLASH on the 64), it cannot find the entry in the lookup table and so will crash. Similarly BASIC 3.5 tokens will fall off the end of the 64's lookup table, and it will also crash.

One further point regarding transfer of programs from one machine to the other, the cassette system used by the Plus/4 and C16 is incompatible with that used in *all* other Commodore computers, so programs can only be transferred through the use of disk media.

Dear Tim,
I am writing to you because I think your magazine seems to have a much more mature point of view to Commodore computing than others such as C+VG and Commodore User.

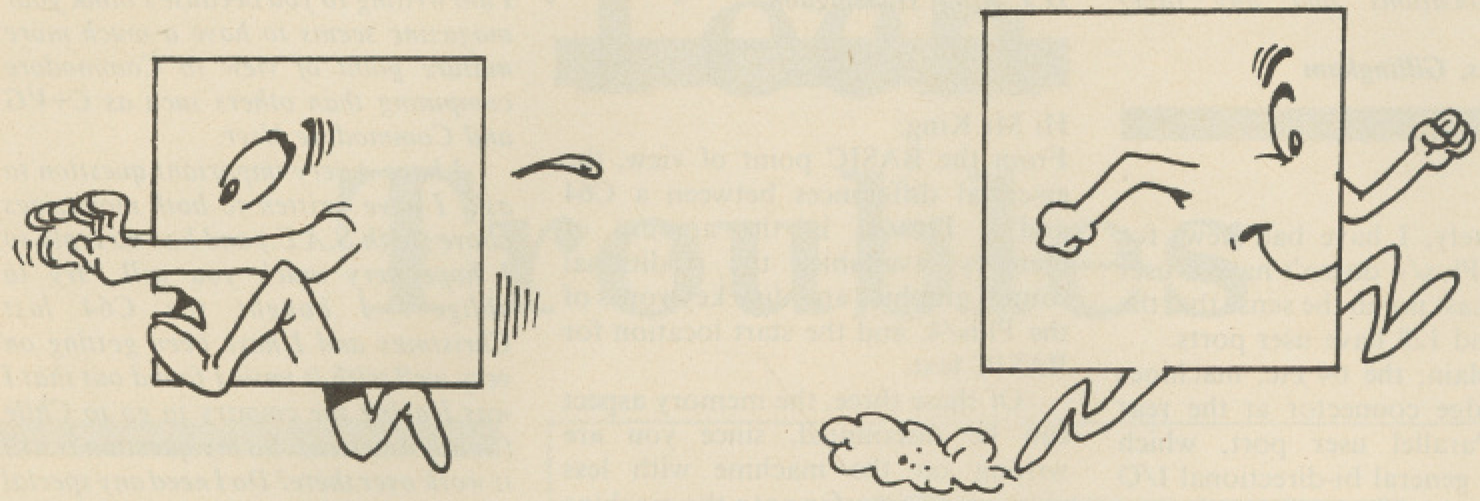
I have a very important question to ask. I have written to both magazines above (with S.A.E.) and had no reply so I hope very much you will "try to oblige". I bought the C64 last Christmas and I have been getting on very well with it until I found out that I was leaving the country to go to Chile (South America). So my question is will it work over there? Do I need any special cables? If so, which? I am using a T.V., will I need a monitor? Will a monitor from over here work on it? Please reply, preferably with good news.

P.S. what do you think of the letter quality? Printed by the "Silver Reed Colour Pen Graph". Could you tell me if there is any compatible software for it to draw graphics?
Rodrigo Araya, Bristol.

Hi Rodrigo,
OK, the easy bit first, you don't need any special cables, neither will you need to use a monitor – a T.V. is quite acceptable, regardless of the country you are in. As for the rest of your question, I had to do some digging. In the UK, and in most of Europe, we use a TV system called P.A.L. which stands for Phase Alternative Line. Additionally our TV system has 625 'scan lines' making up the entire picture. Chile, and indeed most of South America has adopted a different TV system, called N.T.S.C. which stands for North American Television Standards Committee. Their T.V. system has 525 scan lines on the screen. The result is that the two are mutually incompatible, and you will have to sell your 64 system and buy another one when you get there.

The mains voltages in Chile are also different to ours, and so the 64 would not work even if you took your P.A.L. television with you (which could have been an option, although you would have been able to watch local TV on it).

The quality of your printer is quite good. If it is 'Epson compatible', it should be able to run most of the printed graphics packages that are around at the moment.



Sprite Library

More tips on Sprite manipulation on the Commodore 64

by Mike Benn

This month we get out the geometry set. All the sprite definitions are based on single sprites and use hires. The table best illustates the variations on this month's theme with a mixture of animated sequences and single shapes.

GETTING IT ALL IN

Type in the basic loader as published and save it-don't run it or it will self-destruct. Before running the loader program, you'll need to reset the computer and type directly in the following –

```
POKE43,0:POKE44,64:POKE16384,0:NEW
```

– and press return. This will trick the computer into believing that the basic now starts at $4000 instead of $0801.

Load in the basic loader and run it; if error free, the program will automatically save itself as a block of data. If you reload that data in the future, remember to add a 1 after the device number. The data is saved in the following location: $2800-$37FF.

The sprites run from 160 to 223 in a compromise to avoid the area $2000, traditionally set aside for redefined character graphics, and to avoid the need of typing in line after line of data.

If only one or two sprites are required, then use this formula – (sprite block No.-160) *40 + 190 = the data line number at which that sprite blocks data starts. Remember to type in the following three lines of data, and alter the variable BL to the number of data lines you have in your finished program, less 1.

The small basic program GEOM. DISPLAY, will variously animate the sprites in both non expanded and expanded forms on the screen simultaneously. To hold on any sprite, enter the same number for Start and End.

Any sprite Editor program will enable you to change and adapt the individual sprites to your own requirements.

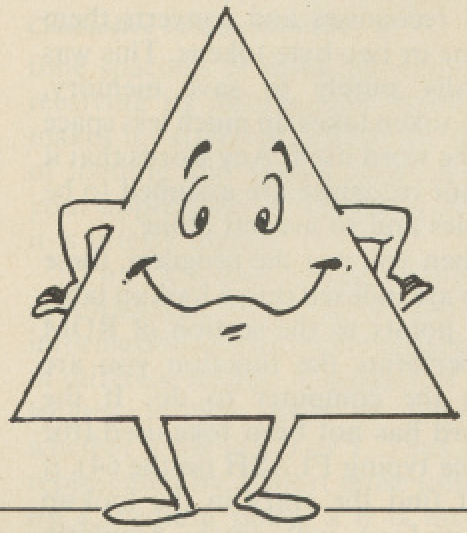


TABLE (Geometric — Hires)

AØ – A3 / 160 – 163 Growing triangle	CF / 207 Inverted pentagon
A3 – A6 / 163 – 166 Spinning triangle	DØ – 04 / 208 – 212 Growing hexagon
A7 – AB / 167 – 171 Growing square	D5 / 213 Turned hexagon
AB – B3 / 171 – 179 Rotating square Y axis	D6 / 214 6 pointed star
B3 – BB / 179 – 187 Rotating square X axis	D7 / 215 Turned six pointed star
BB – BD / 187 – 189 Turning square	D8 / 216 Cube
BD – C3 / 189 – 195 Rotating diamond X axis	D9 – DE / 217 – 222 Growing octagon pattern
C3 – C9 / 195 – 201 Rotating diamond Y axis	DF / 223 Octagon
CA – CE / 202 – 206 Growing pentagon	







Power Play Hockey

What's mean and tough, skates and spends most of his time in the sin bin? The answer is, of course, me playing ice hockey games. They're particularly suited to maniacs like myself who can cheerfully trip and barge opponents while supposedly scoring goals. They usually win the match, but we win the battle.

Seriously, Ice Hockey is an ideal sport to be computerised, and many have tried before and failed to capture all the excitement of one of the world's fastest games. Now Electronic Arts has produced a version featuring superb 3D scrolling graphics but how does it measure up?

For some peculiar reason that's lost on us that live on this side of the pond, the game is based on only one ice hockey match, and is the clash between the good old US of A and the red menace or USSR during the 1980 Olympics, which to nobody's surprise is about the one and only time the Americans won. There's not a hint of Glastnost in this game.

The game can be played in a variety of permutations, from the simple one or two player games and deciding the length of the three periods, to determining the skill level of computer opponents and whether to play one on or five-a-side hockey (that's only counting the out players).

Either way you play, the centre is the man expected to score the goals and save the national honour. You control the centre through standard joystick controls to skate around the rink, but you should remember that 360 degree turns at high speed aren't easy when you're skating on ice, so you have to account for momentum when planning your moves.

Your team meates will play in their positions. In other words, the left and right defense will try and stop the opponent's goal scoring ambitions, and try and win the puck, and the left and right wing help help move the puck forward and to either shoot themselves or pass to you. You can signal for a pass at anytime by pressing the fire button, and they'll do their best to get it to you. Then you can try a wrist shot or slap shot at goal or a drop pass to another player, which is particularly useful if a defender is about to clobber you. However, these moves require tricky joystick manoeuvres and will need some practise before you can defend the honour of the free world.

You have in fact got three squads of players to choose from at the push of a button, they consist of the squad that starts each period who are good all-round players but tire quickly and so you might want to opt for squad two who are real pros when it comes to scoring or the cool in defense squad three. Be warned, that these also tire especially in a 20 minute period so you have to be careful how you play them, or you'll be left two goals down with only the defensive squad fit to play.

Defensive moves are easier to learn and consist of the subtle stick poke by which you try and poke the puck away, the less than subtle slide block to steal the puck and the obvious body check to take out the man. You have to be wary about how you play it otherwise the gloves will come off, a fight will start and you'll spend between one and two minutes in the penalty box.

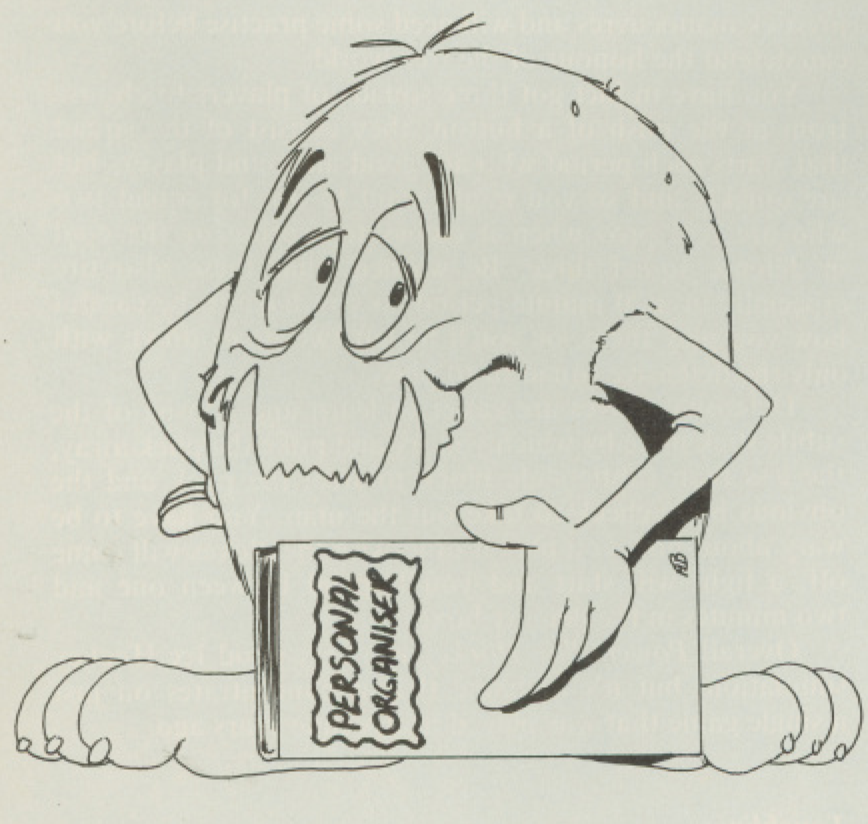
Overall *Power Play Hockey* is a very good Ice Hockey simulation, but it's a shame that it concentrates on just a single game that was played nearly nine years ago.

Touchline:
Title: *Power Play Hockey.* ***Supplier:*** *Electronic Arts, Langley Business Centre, 11-49 Station Road, Langley, Nr Slough, Berks., SL3 8YN.* ***Tel:*** *0753 49442.* ***Price:*** *£14.95 (disk).*

Personal File Database

Avoid elaborate and time-consuming editing procedures with this word processing/database system

By Eric Randall



This utility is a cross between a word processor and a database system. It's aimed at the user who needs to maintain files of text, and be able to readily update and amend the material, without the need to learn and remember elaborate editing procedures. To achieve this, it uses the C64's normal screen editing functions, with one or two more added, and all the instructions needed to use the system are displayed (except when setting up or editing a page of text).

The data is held in pages containing complete screens of information. Each may be recalled and updated as often as required, using the function keys as explained in the main display. New pages may be inserted or old ones deleted, and the entire file in memory can be printed. If, when editing a screen, you need to refresh your memory on which function key to use, press return to save what you've already typed in. The main display will appear and, after checking the instructions, recall the page you were editing by using the F5 key.

Having set up a file of data, it may be saved to disk. To simplify the disk housekeeping, the program takes care of naming files, and you're asked if you want to overwrite an existing file. The file names start with "++data" and end with a number in the range 0 to 9. All the user needs to remember is which number is used for a particular file.

For the technically minded, the data is held as screen codes, not ASCII codes. If you want to extend the scope of this system, this must be taken into account. Furthermore, the data are compacted before filing to disk, so the disk file formats differ from those displayed on the screen. The areas of memory used by the system are as follows:

Basic Program	– $0800	– $1185
Housekeeping data	– $6400	– $64FF
Storage for screen pages	– $6500	– $8BFF
Area for packing and unpacking data prior to storage on disk	– $8C00	– $9FFF
Machine	– $C000	– $C4B0

For those wishing to experiment with and extend the system, the entry points to various machine code routines called from the main BASIC program may be useful. Most of these routines require parameters which are taken from various locations between $6400 to $64FF. These are set up in the main program, and the references are as follows:-

LOADING FILES

Lines 300-350 SYS49320 ($C0A8) – load a file
SYS50248 ($C448) – unpack data

SAVING FILES

Lines 90-210 SYS50093 ($C3AD) – pack data
SYS49299 ($C093) – save a file
SYS49348 ($C0C4) – overwrite existing file

CREATING NEW PAGE OF DATA

Lines 500-555 SYS49510 ($C166) – turn on interupt routine to scan for function keys which tabulate up and down and which insert and delete lines.
SYS49535 ($C17F) – turn off interupt routine
SYS49152 ($C000) – store screen

UPDATE PAGE

Lines 700-720 SYS49243 ($C05B) – display stored page
SYS49510 ($C166) – turn on interupt routine
SYS49535 ($X17F) – turn off interupt routine

PRINT FILE IN MEMORY

Lines 800-840 SYS49358 ($C0CE) – Print file

DELETE A PAGE

Lines 600-605 SYS49916 ($C2FC) – delete a page

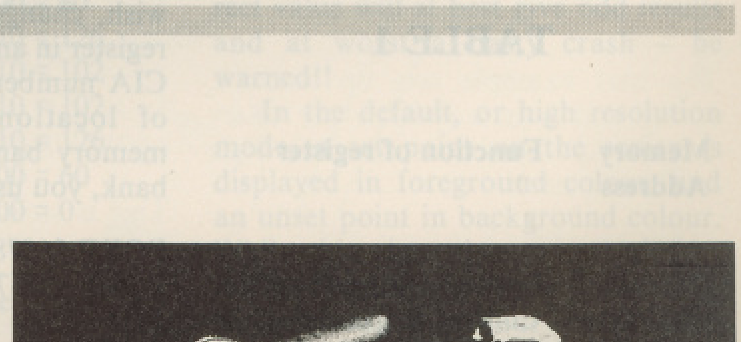
INSERT A PAGE

Lines 650-655 SYS50000 ($C350) – insert page

USING THE SYSTEM

To start you off using the system, a data file is included on the disk as file number 0. This contains the headings of personal information for you to fill in the details.

Introduction to Graphics on the



by Allen Webb

The first of a two-part series examining the considerable graphics potential of the Commodore 64

For many new owners of the Commodore 64, playing games soon becomes inadequate, and they turn instead to programming and trying to emulate the graphics effects that the game writers achieve. The machine's manual will help you a little, but it makes no attempt to get to grips with the not inconsiderable graphics capabilities of the machines. In this series, I'll give you a taste of what you can achieve with Basic. You must realise that using Basic will limit your potential, but you will nontheless be able to get some excellent results.

The heart of the graphics power of the 64 is the VIC chip, which has a collection of memory – mapped registers which handle graphics. It's a rather sad fact of life that Basic doesn't support graphics in any way. Instead, you'll have to learn a collection of POKE commands to manipulate the VIC registers. Table 1 gives a summary of the registers and their functions.

TABLE 1

Memory Address	Function of register
53248-53263	Sprite position registers
53265	Control Register 1
53269	Sprite display enable
53270	Control Register 2
53271	Sprite Vertical Expand
53272	Memory Control Register
53273	Interrupt Flag Register
53275	Sprite Priority Register
53276	Sprite colour mode
53277	Sprite horizontal expand
53278	Sprite collision detect
553279	Sprite collision detect
53280	Border colour
53281	Screen colour
53282-53284	Background colours
53285-53286	Sprite multicolours
53287-53294	Sprite colours

In these articles I will explain how to tweak these registers to good effect.

First – I must discuss how the machine manages memory. The Commodore 64, as indicated by its name, has 64 kilobytes of memory. This can be imagined as a sequence of 65536 boxes with can obtain data. EAch box (or memory location) is numbered, starting at 0 and continuing to 65535. The number of the box is called its address.

The machine uses various chunks of memory for various purposes. Due to its ROM – based software, the 64 uses the same areas of memory for its various functions. You can, however, change things around if you wish. The VIC chip can address only 16K of memory at any one time. The 64 has four "banks" of 16K available for use, and these are numbered as shown in Table 2.

TABLE 2

Bank Number	Block of memory	Z
0	0 - 16383	3
1	16384 - 32767	2
2	32768 - 49151	1
3	49152 - 65535	0

When you turn on the 64, bank 0 is automatically selected. This means that the VIC chip expects to find the screen memory and the character set designs in this area. You can, if you wish, change the bank by altering a register in another important chip, the CIA number 2. The bottom two bits of location 56576 determine the memory bank in use. To change the bank, you use the following two lines:

```
POKE 56578, PEEK(56578) OR 3
POKE 56576, (PEEK(56576) AND
252) OR Z
```

Using some banks will give a screen full of garbage. I will tackle this feature fully next time, so forget all about it for now.

The display uses a block of memory to hold its contents. The display consists of 25 lines of 40 characters, requiring 1000 bytes. In the default 64, the screen memory starts at memory location 1024 and continues to location 2023. In order to provide colours, each screen location has a corresponding colour memory location. This occupies the 1000 bytes starting at 55296. The colour memory is always at this address. Try the following program to see how it works:

```
10 FOR I = 0 TO 999
20 POKE 1024+I,1
30 POKE 55296+I,RND(1)*16
40 NEXT I
```

The machine alsoneeds to know what shape the characters are, and this information must be held in the active memory bank as well. Because the 64 is an 8 – bit machine, it allows a maximum of 256 characters. Each character requires 8 bytes of memory to hold its shape data. Hence, the full character set requires 2048 bytes. Since there are two full character sets available, a total of 4096 bytes are used. In the default system, the character set is held in a Read Only Memory, and is found in the block 4096 to 8191. Because this data is held in this way, it occupies the same space as Random Access Memory, so that you don't lose any memory. Given these facts, Table 3 shows a general memory map:

TABLE 3

Memory Address	Use
0 - 1023	Used by system
1024 - 2023	Screen memory
2040 - 2047	Sprite pattern pointers
2048 - 40959	Used for BASIC programs
53248 - 53294	VIC registers
55296 - 56295	Colour memory

The parts of memory not described are occupied by the system ROMs and other chips. The position of the screen memory and the character set are determined by the contents of the memory control register in the VIC chip (53272). More on that in a moment.

When you put a value in a screen memory location, the system translates this value into a specific character pattern in the character memory. A value of 0, for example, uses the first pattern (or group of eight bytes). This pattern represents @ – value of one uses the second pattern and so on. The following program shows the full character set by POKEing the full range of values into the screen memory:

```
10 FOR I = 0 TO 255
20 POKE 1024+I,I
30 NEXT I
```

The next step is to place a character at a specific place on the screen. If R is the row and C the column, the

correct address is given by:

1024+(R-1)*40+(C-1)

Where R is in the range one to 25, and C is in the range one to 40.

The following example uses this relationship as a function to move an asterisk diagonally across the screen:

```
10 DEF FNA (R)=1024+(R-1)*40+(C-1)
20 FOR R=1 TO 20
30 C=R
40 POKE FNA (R),42: FOR D=1 TO 30: NEXT D
50 POKE FNA (R),32: FOR D=1 TO 30: NEXT D-
60 NEXT R
```

This routine first puts on asterisk at each point (character 42) followed by a space (character 32). The two loops in lines 40 and 50 are simply delays to give a better effect. While making some progress, this effect is a little boring. What's needed are custom-designed characters, but how can we achieve this? As I have already described, the normal character set is etched in a ROM, and what we want is a character set in RAM. The secret is to redirect the pointer to the character set to a handy bit of unused RAM. The memory control register allows this. The bottom four bits of this register decide the offset position of the character set in any given bank. Take a look at Table 4.

TABLE 4

Memory Offset	Z
0	0
2048	2
4096	4
6144	6
8192	8
10240	10
12288	12
14336	14

The character set pointer is changed by using:

POKE 53272, (PEEK(53272)AND240)OR Z

To find the start address of the character set, you simply add the offset to the start address of the bank (given in table 2). Let us consider how a character is designed. Here is a sample character:

```
...**...   00011000 = 24
..****..   00111100 = 60
.**..**.   01100110 = 102
.******.   01111110 = 126
.**..**.   01100110 = 102
.**..**.   01100110 = 102
.**..**.   01100110 = 102
........   00000000 = 0
```

The stars represent visible dots, and the dots represent spaces. Each pattern can be converted to an eight bit binary number by replacing stars by 1 and dots by 0. These numbers are then converted to decimal, and the resulting eight numbers are those used in the pattern table. Here is an example of a redefined character:

```
........   00000000 = 0
..****..   00111100 = 60
.******.   01111110 = 126
.**..**.   01100110 = 102
.**..**.   01100110 = 102
.******.   01111110 = 126
..****..   00111100 = 60
........   00000000 = 0
```

The following example uses this character plus one other to produce stunning animation...Gasp!

```
10 POKE 53272, (PEEK(53272) AND240) OR 14
20 DATA 0,60,126,102,102,126,60,0
30 DATA 0,0,0,24,0,0,0
40 FOR I=0 TO 15: READ X: POKE 14336+I,X: NEXT I
50 FOR I=0 TO 7: POKE 14336+32*8+I,0: NEXT I
60 DEF FNA (R)=1024+(R-1)*40+(C-1)
70 FOR R=1 TO 20: C=R
80 POKE FNA(R),0: GOSUB 200
90 POKE FNA(R),1: GOSUB 200
100 POKE FNA(R),32: GOSUB 200
110 NEXT R
120 END
200 FOR D=1 TO 30: NEXT D: RETURN
```

In this example, I have placed the start of the new character set at 14336. Line 40 reads the new character data and puts it into the character data table. Line 20 holds the design described earlier. Line 50 sets the space character (32) to a space. The rest of the characters are garbage, since we haven't set them to anything interesting. If you want to return to the normal character set, the quick and dirty way is to press RUN/STOP and RESTORE together.

The normal display comprises two colours, the background and the foreground – in Sinclair Spectrum terms, paper and ink – and some border colour. These colours can be easily altered by putting the colour value (0 to 15) into the relevent register. The registers are:

Border	53280
Background	53281
Foreground	646

Since these registers use only the bottom 4 bits of the stored value, any value greater than 15 will simply result in cycling the colour sequence. In general, you should ensure that the value stored in any VIC register is accurately controlled since an incorrect value will at best give odd results and at worst a nasty crash – be warned!!

In the default, or high resolution mode, a set point on the screen is displayed in foreground colour, and an unset point in background colour. While this gives the greatest clarity, it is not best suited to pretty graphics. So we enter multicolour mode, which can give you up to four colours in any one character. The penalty is that the horizontal resolution is reduced from eight dots per character to four. The reason for this is that each pair of dots in the character pattern determines which colour is used. The colours are obtained as shown in Table 5:

TABLE 5

Colour Source	Bit Pair
background 53281	00
background 1 53282	01
background 2 53283	10
low 3 bits of colour RAM.	11

Let us consider one byte of a pattern:

...**.** = 00011011

In high resolution mode, this would look like a pair of dots. If we split it into four pairs of bits we get another image:

00011011 = 00 01 10 11

Using Table 5, we can see that in multicolour mode, the first pair would appear as the background colour, the second pair would appear as a dot coloured in background one, the third as background two and the last as the colour ram. Because only the bottom 3 bits of the colour ram is used, this bit pair can only use the first eight colours. The other bit is used to toggle multicolour mode for the particular character. You see, if the fourth bit it set, the character is set to multicolour, or else it is set to high resolution. In practical terms, you simply add eight to the colour value (in the range nought to seven). This allows you to mix both modes on screen. To turn on multicolour mode you use:

POKE 53270, PEEK(53270) OR 16

and to turn it off:

POKE 53270, PEEK(53270) AND 239

The next example sets up a black screen and puts a multicolour character next to the same character in high resolution mode.

```
10 POKE 53280,0: POKE 53281,0
20 POKE 53282,6: POKE 53283,3
30 POKE 53270, PEEK(53270) OR 16
40 POKE 1024,1: POKE 1025,1
50 POKE 55296,1: POKE 55297,8
```

The lines 10 and 20 set up the colour registers. Line 30 turns on multicolour mode and line 40 puts an A in the top left two screen positions. The first POKE in line 50 clears the fourth bit of that particular colour memory location, thereby setting the ink colour to white, and setting the character to high resolution. The second POKE sets bit 4, thereby setting the ink colour to white and setting the character to multicolour.

Trying to design multicolour characters "by hand" is both tedious and difficult – if you want to get good results, it would be worth your while investing in a good quality character designer.

If all of this is not enough, there is one more character mode – extended background mode. In this mode you have a high resolution character, but with a choice of four background colours. This time the system gets the colour choice from the screen contents. In simple terms, the background colour depends on the character on the screen, and the foreground colour depends on the colour memory. Table 6 shows how:

TABLE 6

Colour Register	Character range
53281	0-63
53282	64-127
53283	128-191
53284	192-255

To turn on extended background you use:

POKE 53265, PEEK(53265) OR 64

and to turn it off:

POKE 53265, PEEK(53265) AND 191

Here is a simple example:

```
10 POKE 53280,0: POKE 53281,0:
POKE 646,1: PRINT CHR$(147)
20 POKE 53284,11: POKE 53282,12:
POKE53283,15
30 POKE 53265, PEEK(53265) OR 64
40 FOR I=0 TO 255
50 POKE 1024+I,1
60 NEXT I
```

This is simply a repeat of an earlier program. Here I have used PRINT CHR$(147) to clear the screen. On the newer 64s this fills the colour RAM with the current foreground colour. By the way don't try to use extended background mode with multicolour mode – it won't work.

As I mentioned earlier, it's possible to relocate the block of memory used for the screen display by altering the relevent bits in the memory control register (53272). This time, it's bits 4 to 7. For most users, this option is not of great use, but it does offer the option of having several screens active at once, with a simple means of switching between them. This could allow animation sequences, and other effects.

In a manner analogous to the character set, the screen can occupy a range of positions. This time, in view of its lesser size, the screen can be placed in one of 16 positions. These positions are in increments of 16.

Table 7 shows the available options:

TABLE 7

Location of Screen (offset from start of bank)	Z
0	0
1024	16
2048	32
3072	48
4096	64
5120	80
6144	96
7168	112
8192	128
9216	144
10240	160
11264	176
12288	192
13312	208
14336	224
15360	240

The first step is to change the pointer to the screen with:

POKE 53272,(PEEK(53272) AND 15) OR Z

As before, the start address offset is added to the start address of the current bank to get the absolute screen address. We if you intend to use BASIC, you must also tell the editor where the screen is. A pointer to the page occupied by the screen is held in location 648. This must be altered. You get the page by obtaining the absolute address of the start of screen memory (as described in the last paragraph) and dividing it by 256. The 6502 micro-processor handles memory in pages of 256 bytes. The result is POKEed into location 648.

I must apologise to beginners for the large amount of theory in this part. Unfortunately it is unavoidable, and in any event, I want this short series to become a simple reference guide to which you can refer when necessary.

Well, that's about it for now. With the building blocks developed here, you should be able to start developing some interesting graphics of your own. Next time, I'll be looking at bit-mapped graphics and sprites. I will also include a small package of machine code routines which will ease the use of the VIC chip.

Relative File Programming

In the last of his series, Eric Ramsay gives some expert advice on filing with Commodores

To conclude this series, I have included every routine, including those for Basic 2 as well as the easier Basic 7, that is necessary for you to program your own files in Basic.

I'd also like to make a few suggestions which you might consider for your own applications. For example, where you have many thousands of records to access, you would of course have the Keyfield Search Routine I've already shown you. This is quite fast – being able to find, in my case, a match in one record in a file of over 1,000 in less than nine accesses – but there is another way: an index file.

An index file contains details, not of all your records (although it could, this would be very wasteful), but of selected records. As you may recall, the search routine included a routine which repeatedly halved the numbers of records until it arrived at the one searched for: a method similar to this could be used in order to write an 'access' file. For example, imagine you have 4000 records already created, that have already been sorted. These could be sectioned, a sample record at each section read, and the keyfield found there saved in a file. This would be something like this:

```
5000 a(0)=int(rn/2):forq=1to5:a(q)
=int(a(q-1)*2):nextq
```

This loop has effectively sectioned the file, so that using them you may take a sample of the sorted keyfield at that point.

Once having run this routine, a separate index file will exist which you can use to narrow down the search of a particular string within the keyfield by using the same loop:

```
5010 dopen#4,(left$("index"+
name$+"......",16))
5020 dopen#5,(name$),d0,u8:
gosub3610
5030 forq=1to5:rem start loop
5040 rrec=rsrt%(a(q)):gosub
4120:rem Read Record
5050 gosub4180:print#4,
disp$(kf)
5060 gosub 3610
5070 next q
5080 dclose

6000 a(0)=int(rn/2):forq=1to5:a(
q)=int(a(q-1)/2):nextq
6010 dopen#4,(left$("index"+name
$+"......",16)),d0,u8,r:gosub361
0
6020 forq=1to5:input#4,comp$
6030 ifse$<comp$thenbeg=q:q=5:dc
lose:return
6040 next q
```

At this point, the loop to read in the comparison sample string has been opened, and is then compared with the search string inputted previously. If the search string is less than that of the comparison string, then obviously if a match is to be found it must be within the previous sample range. A(A-1) stores the record number for the previous range of sample, so the beginning variable is set to that. The limit is set:

FOR PO = A(beg) TO A(BEG-1):RREC=RSRT%(PO)

and the read routine can be called. This facility automatically limits a sequential scan to exactly the range where the record match is to be found. In a file of 4000, this would reduce a search to 65 records.

Sequential data access method

If you're planning a relative file with a fixed keyfield, you may care to try the sequential data access method. This does not require a sequential file as an index, but several relative files, all inter-related. This is suitable only for a static file, one which is not continually being amended, added to or deleted from. The principle is as follows:

The keyfield only is stored, together with its sorted record number in a separate relative file according to its alpha or numeric value. Thus, a record with the keyfield of Smith John would be filed, in alphabetical order, in a relative file for keyfields beginning with S. A short routine would detect the first letter of the search string, either by its ASCII value or by comparison inside a string, and then reading, not the full relative file, but the relative index file. The index file would be opened, and each record scanned until the match was made. In this case, the program would read the record:

SMITH John 3129

The number 3129 would direct the program to precisely the correct record number to access.

More than 256 bytes per record

Impossible, you might say. Well, yes and no. There's no reason why you should limit yourself to the DOS maximum of 256 bytes per record. You might like to try using a record as one field of a database record. Confused? Well, imagine you wanted six fields per record, each of 80 characters. This would clearly be outside the DOS maximum record size – 480 bytes.

But if you were to treat each of the disk records as a field, then you could have 2088 fields in your file. There would be six of these fields in each of your file records, so you would have a maximum of 348 file records for use. Searching a particular field would be a doddle. To read the first field of your file record would mean looking at the first seventh, 14th, 21st record and so on.

Taking that a stage further, you might wish to consider having a separate relative file for each field in your file record. Your total file length could then be the total formatted capacity of your particular disk drive divided by the number of fields you wished to use. In this way, every byte of your floppy would be used.

Taking that a stage further, if you used two disk drives, you could place half of your file record on one drive and the other half on the second. This would mean a total file capacity of 349,696 bytes on two 1541, or 1570 drives, and 699,392 on two 1571 drives. Now that would be a challenge!

Mailbag

Your chance to have your say



When I look at my 128 and my Amiga I am reminded strongly of the old fable of the Grasshopper and the Ant. The Grasshopper played all summer long, leaping and gambolling all over the meadow, showing off, but never doing anything really useful. The Ant on the other hand toiled steadily away, doing all the real work and laying up stores. When winter came the poor old Grasshopper had no reserves so he perished. At least the Amiga has made it through the winter, and done enough in the video field to make sure it will prosper, but while it may remain the machine for the future, I maintain that the 128 is that of the here and now.

I am not knocking the Amiga. I think it is a wonderful machine. If your main interest in in video graphics then the Amiga is your only choice. It is what I had hoped the C64 would be like when I bought it nearly five years ago. I remember, however, tramping around the computer shops trying to find a decent program, and am amazed at the things that are now being done with a C64. Even the designers hadn't any idea how powerful it would prove to be. The Amiga is in the position now that the C64 was five years ago – full of promise and potential, but with no really great programs around to exploit it. I only hope I won't have to wait another five years for the programmers to get to know the Amiga.

I started with a C64, but hankered after an 80-column screen, so I took the plunge and bought a Commodore 128 and high-resolution colour monitor. I have never regretted it. Despite the scarcity of programs written for 128 mode there were enough to make the 128 a real work-horse, and in any case one could revert to C64 mode to play *Elite, Leaderboard World Class,* Chess or Bridge.

But always there was the sheer usefulness of the 128. I took on my wife's correspondence as secretary of a charitable organisation, did *Deanery Dates* for the vicar, and later produced professional-looking posters for both of them. Later I ventured into writing simple programmes in the enhanced Basic that can be found as the programming language of Superbase. It is this tireless, accurate, time-saving, data-gobbling aspect of home computing which fascinates me.

When the Amiga first arrived it was priced out of my reach. I could only envy those who could afford one. I think the majority of 64/128 owners felt the same. Then came the A500, and I decided that I ought to keep up with the times. I chose the relatively cheap way in by buying an A500 and having my 1901 Monitor converted by Trilogic of Bradford for only £25. It equals the dedicated Amiga Monitor for colour and definition, but it can, of course, still be used with the C128.

I must confess that I was almost drooling over the new possibilities – a seemingly endless memory, and a very speedy disk drive. New technology would open the flood-gates and things would really take off. True, there weren't many programs available, and even games were twice the price, but time would cure that. The instantly noticeable difference was the graphics. No more clumsy blocked sprites. No more peering into a blurred screen trying to guess which chunk of blocks was the alien. I bought *Defender of the Crown* and marvelled at the incredible detail. Then I took a flight round the Statue of Liberty and the rot set in. Instead of the rounded lady of the C64 (albeit floating past the side windows rather jerkily) I could see only a cardboard cut-out. I may be wrong. Maybe my copy has a glitch. But the first faint shadow of doubt had crept in. The sun was beginning to cloud over a little. I returned to *Defender of the Crown*. The scenery was as delightful as ever, but was the game-play a little stilted? Even boring? Yes it was. Very! I bought *Leaderboard* for the Amiga. Quite frankly I preferred the multiplicity of courses on the C64. And it played just as well as on the Amiga. *Star Glider* was quite good, but I prefer *Elite* on the 64.

Ah, yes, but it must be much better for the really productive work? Well, I have tried *Scribble, Vizawrite Desktop* and finally *Wordperfect*, but for a really user-friendly yet very powerful word-processor, which can be customised with a built in language, give me *Superscript 128* in 80 columns on the C128 any time. Yes I know that familiarity plays a part, but a new, powerful 16-bit program should overcome that, surely.

I will continue to work with the Ant, while playing with the Grasshopper, waiting impatiently for it to get down to some real work. With 128s available second-hand for £125 (try Once Bytten) and lovely printers like the Panasonic 1081 giving very near letter quality as cheap as £150 new (with cable), it seems that the Ant has a lot of hard work in it yet. New programs and utilities are pouring in from America, and I gather that the new 1581 3.5-inch disk drive is fast and reliable (six times the speed and four or five times the capacity of the old 1541, so I'm told). Most of the programs available are tried and tested. Spreadsheets, databases, wordprocessors and the like abound, all at prices far lower than similar material for the Amiga and PCs.

So don't sell the C64 or 128 to buy an Amiga. Save up a little longer, and buy an Amiga as well, and while you are waiting continue the good work with the 64 or 128!

Routine Matters

Moving ROM into RAM is too slow in Basic – but there is a quicker way

By Eric Doyle

Routine Matters

PROGRAM: MEMORY MOVER

```
91  10 REM TYPE THIS IN AND SAVE
     IT
22  20 REM TYPE NEW BUT BEFORE R
    ELOADING TYPE IN POKE43,1:PO
    KE44,64:POKE16384,0:NEW
AB  30 REM THEN LOAD THE PROGRAM
     USING ,8 OR ,1 - NOT ,8,1 O
    R ,1,1
F2  40 OS=53248:OE=57344:NE=1638
    4
DC  50 Z1=NE:GOSUB60005:Z6=Z2:Z5
    =Z1
91  60 Z1=OE:GOSUB60005:Z4=Z2:Z3
    =Z1
95  70 Z1=OS:GOSUB60005
9A  80 PRINT"[CLR]"
C1  90 FORA=0TO255:POKE1344+A,A:
    NEXT
1C  100 POKE56334,PEEK(56334)AND
     254
3B  110 POKE1,PEEK(1)AND251
AF  120 GOSUB60015
F8  130 POKE1,PEEK(1)OR4
15  140 POKE56334,PEEK(56334)OR1
09  150 POKE53272,(PEEK(53272)AN
    D240)+12
B6  160 PRINTCHR$(142)
EE  170 FORA=12288TO12288+4095:P
    OKEA,255-PEEK(A):IFA=14335TH
    ENPRINTCHR$(14)
2E  180 NEXT:IFB>999THENPRINTCHR
    $(142):END
8D  190 FORB=1TO1000:NEXT:GOTO16
    0
47  200 REM
5D  210 REM
53  220 REM
80  60000 REM * BYTE SPLITTER *
    Z1=BYTE TO BE SPLIT
A2  60005 Z2=INT(Z1/256):Z1=Z1-Z
    2*256:RETURN
DF  60006 REM
DE  60007 REM
97  60010 REM * MEMORY MOVER * Z
    1=LO OLD START:Z2=HI OLD STA
    RT:Z3=LO OLD END
17  60011 REM Z4=HI OLD END:Z5=L
    O NEW END:Z6=HI NEW END
AA  60015 POKE678,165:POKE679,25
    1:POKE680,133:POKE681,95:POK
    E682,165:POKE683,252
CF  60020 POKE684,133:POKE685,96
    :POKE686,76:POKE687,191:POKE
    688,163
B3  60025 POKE251,Z1:POKE252,Z2
5E  60030 POKE90,Z3:POKE91,Z4
FF  60035 POKE88,Z5:POKE89,Z6
4D  60040 SYS678
F9  60045 RETURN
```

It's not always that a whole character set has to be redefined, but moving ROM into RAM takes 70 seconds. Machine code is much faster, and a memory mover is included in the Basic ROM, but the problem is how to access it.

The memory move is called by several routines in the ROM. Principally it's called when a line of Basic has to be inserted into a program. A new line may be added anywhere in the program, and a space has to be made to allow this. This may mean that a few bytes or a few thousand bytes may have to be moved, so the routine has to be very pretty awesome.

With no changes, the routine can be harnessed to move memory about quite happily, as long as the start and end addresses of the block to be moved are defined alongside the *end* locations of the new location. The two end locations are the actual end locations plus one byte.

First of all, the subroutine will need to know these locations, so another subroutine is required that will split the high and low bytes of the three locations needed. Using a local variable, Z1, the three bytes can be dealt with separately by this routine.

```
60000 REM * BYTE SPLITTER *
Z1=BYTE TO BE SPLIT
60005 Z2=INT(Z1/256):Z1=Z1-Z
2*256:RETURN
```

The ROM characters are stored from 53248 to 57344, so these two values are fixed. It only remains to decide where the characters are going to be stored. The screen is located at 1024, so the redefinable characters will have to be located somewhere in the first 16K. By raising Basic start to 16384, the start of the character set could be 12288. 4096 (57344-53248) bytes are going to be moved, so the end location is 12288+4096 or 16384!

Moving Experience

The byte splitter routine returns the high and low byte values of each location as Z2 and ZZ1 respectively so, after each location has been split, the program must reasign these variables as something else. Our block moving routine will use four variables from this process so Z6 to Z1 can be used:

```
60010 REM * MEMORY MOVER * Z
1=LO OLD START:Z2=HI OLD STA
RT:Z3=LO OLD END
60011 REM Z4=HI OLD END:Z5=L
O NEW END:Z6=HI NEW END
```

The ROM routine takes the start location from locations 95 and 96 (lo-hi format), the end location is stored in 90 and 91, and the new end location is expected to be in 88 and 89. This would be a simple case of poking the calculated values into these locations, except for one thing – locations 95 and 96 are frequently used by Basic, and would be cleared by the POKE routine itself.

The only way round this is to set up a very short machine code routine which stores these two values and then jumps into the ROM routine at 41919. This is done by storing the high and low bytes in the free zero page locations 251 and 252 before the routine is called.

First the code routine is installed, then the memory locations are filled with the correct values, and the code is executed with an SYS call:

```
60015 POKE678,165:POKE679,25
1:POKE680,133:POKE681,95:POK
E682,165:POKE683,252
60020 POKE684,133:POKE685,96
:POKE686,76:POKE687,191:POKE
688,163
60025 POKE251,21:POKE252,22
60030 POKE90,23:POKE91,24
60035 POKE88,25:POKE89,26
60040 SYS678
60045 RETURN
```

The machine code is entered as a series of pokes, because a loop which read data in could cause problems with data recessing whenever this routine was called, especially if there are data statements in the main program.

To demonstrate the use of the program, there is a routine in the Listings pages at the back of the magazine which demonstrates how to move the character block into RAM, and then redefines all 256 characters as reversed versions of their original forms. After a short pause, this process is reversed to restore the characters to their original appearance.

The loss of 14335 bytes of memory may seem like a large sacrifice, but the spare memory below 12288 could be used to store sprite date for sprites 32 to 191. Alternatively, the screen could be moved higher up the memory, and the character set stored alongside it. This would mean a fairly complex set-up routine, however.

Another use for this memory space would be to store a second character set. The character set in use could be moved to 2048 and the second character set could be moved from 6144 to 12288. In this way, the two sets could easily be swapped at will.

Apply Yourself

ROM moving is not the only application for this routine. Another exercise would be to use the routine in a program based on pull-down or pop-up menus. Even though the menu may only overwrite three lines of the screen to a width of maybe ony six or seven characters, the move subroutine could be used to store all three complete lines in the memory are at 49152. After the menu selection is made, the same routine could restore the three lines instantly.

I'm sure that you will come up with many more ingenious applications, but remember that the memory mover is a one way device. For example, a block of memory could be moved to start at 2049 if its original start was 2048, but it couldn't be moved to 2048 if it started at 2049. The reason is that the memory moves work from the end to the start. Imagine that the block starts at 2049 and ends at 2100. If a move is attempted to cover 2048 to 2099, what would happen?

First of all, the routine would pick up the content of 2100 and store it in 2099. Next, the content of 2099, which has just been overwritten, would be moved to 2098 and so on. The result would be a block from 2048 to 2100 which would only have one byte, that which was in 2100 originally! For safety's sake, make sure that any block moves don't access locations which the block to be moved already occupies. You can jump either way, but if you're going to jump, make it a big one.

First Steps

The C128, C16 and Plus4 all have their own sets of strange error messages

By Norman Doyle

The extended Basics of the newer Commodore computers mean that extra error messages are required which don't feature on the C64. The C16/Plus4 machines added another six, and this was built-on with the advent of the C128 by a further five.

Can't Resume
Occurrence: programming error

The TRAP statement acts like a GOSUB command, but acts globally like an interrupt routine. Whenever any error occurs, the program jumps to the error handling subroutine situated at whatever line number follows TRAP. The subroutine is terminated by RESUME, which acts like a RETURN in a GOSUB routine. If a program is written with the error-handling routine at the end and no END command is encountered, execution will run into the error routine, and when RESUME is encountered this error message will be generated.

Loop Not Found
Occurrence: programming error

This means that a DO command has been encountered, but the program cannot find a corresponding LOOP command.

Loop Without Do
Occurrence: programming error

This is the opposite of the previous error. It occurs when a LOOP command is encountered but a DO command isn't currently active.

Direct Mode Only
Occurrence: user error

Some of the extended Basic commands, such as AUTO, can only be used in direct mode. In other words, they must be typed onto the screen, and then the RETURN key must be pressed. Any attempt to include such a command in a program will be met with this message.

No Graphics Area
Occurrence: programming error

This means that the HEADER command was used without an ID to rename a disk without going through the lengthy procedure of formatting. Unfortunately, the disk chosen hasn't yet been formatted, so the solution is to repeat the HEADER command with an ID added.

Bad Disk
Occurrence: carelessness or bad luck

At best, this means that the HEADER command was used without an ID to rename a disk without going through the lengthy procedure of formatting. Unfortunately, the disk chosen hasn't yet been formatted, so the solution is to repeat the HEADER command with an ID added.

At worst, it means what it says – the disk is faulty. This can be caused through rough or careless handling, but occasionally it means that the disk was faulty in manufacture or simply worn out through over-use. Throw the disk away and start again with a fresh one.

The next group of errors are the ones which only appear on the C128.

Bend Not Found
Occurrence: programming error

When an IF...THEN BEGIN or IF...THEN...ELSE BEGIN construct is found, the computer then looks for the corresponding BEND command. If it can't find one, it flags this error.

Line Too Large
Occurrence; rare

The highest possible value for a program line is 63999. If an attempt is made to RENUMBER a program and any of the renumbered line values will exceed 63999, the command is aborted without being executed, and this error message is displayed.

Unresolved Reference
Occurrence: program error

If an attempt is made to renumber a program and the programmer has erased a line which is referenced by a GOTO, GOSUB, ON or TRAP command, the RENUMBER routine will not be able to execute properly. Once again the command is aborted without any action being performed, but this time the unresolved reference message is returned.

Unimplemented Command
Occurrence: program error

Two commands are included in C128 Basic, but have no action associated with them. These commands are QUIT and OFF. If either is used, the computer politely reminds the user that it is an unimplemented command.

File Read
Occurrence: operational error

This means that a file or program has failed to load. The reasons could be because the disk has been damaged or because an over-anxious user has opened the drive door before the program has finished loading.

Contributions

So you own a Commodore? So you've written some programs? So why haven't you sent them to us?

Your Commodore is always on the look out for new programs, hints and tips, articles and even regular series. In fact if you have something that you think could be of use to other Commodore owners we want to hear about it.

So if you have got something which you think we may be interested in. How do you go about submitting it to us?

Below you will find a list of guidelines that will help us to deal with any item that you send in to us. We don't expect everybody to be the next William Shakespeare but if you do follow these simple rules then it will make our job a lot easier.

1) If possible all material sent to the magazine should be typed or printed out on a computer printer.

2) All text should be double spaced i.e. there should be a blank line between each line of text. You should also leave a margin of about 10 characters around the text.

3) On the very first page you should put the following:
- Name of the article
- Machine that it is for
- Any extras required – disk, printer etc.
- Your name
- Your address
- Your telephone number

4) The top of every page should have the following information on it:
- Abbreviation of the article title
- Your name
- The page number

For example, suppose you had submitted an article on C64 interrupts. You should put something like the following at the head of the page:

Interrupts/J.Smith/1

5) Please make sure that you do not make any additional marks on your text especially underlining.

6) Try and write in clear concise English, it does not have to be a work of literature but it must be comprehensible.

7) On the bottom of each page you should put the word MORE if there are more pages to the article or ENDS if it is the last page.

8) If possible, enclose a listing of all programs.

9) Under no circumstances use a staple to hold the pages together. Use a paperclip instead.

10) Programs should be included on either disk or tape. Make sure that you SAVE two copies of every program so that we have a better chance of loading them if problems occur.

11) Programs under 10 lines can be included in the text. If your program is longer than this you must enclose a disk or cassette.

12) If your article needs any artwork then supply clear examples of what is needed. We don't expect you to be an artist but we do need to see what is required.

13) Photographs, if necessary, must be either black and white prints or colour slides. We can take shots ourselves so don't worry about this too much.

14) Submissions of any length are welcome. If you have a five line routine that you think may be of use to someone else we welcome it just as much as a full blown six part series.

15) Payment varies quite a lot and depends on quite a number of factors, such as complexity of program, presentation of program, number of magazine pages it takes up etc. Payment is generally between £10.00 and £800.00.

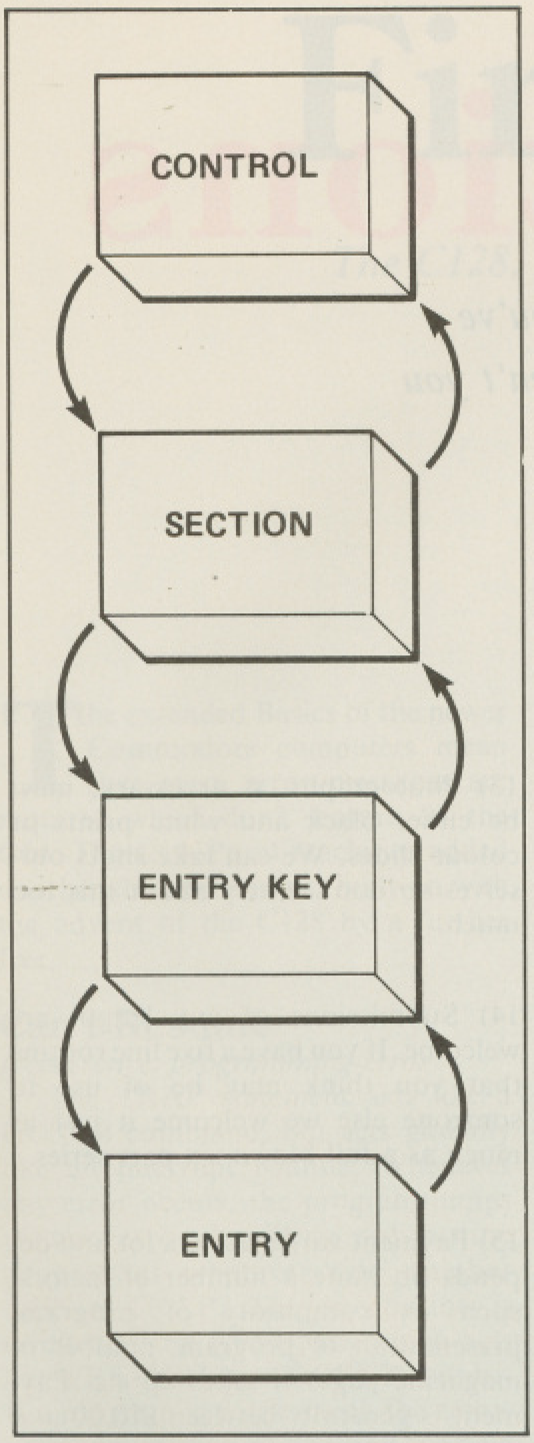
16) All payments are made in the month that the magazine containing your article has appeared in print.

17) If we do find your submission suitable for inclusion in the magazine we will write to you giving the terms of publication, the rate of payment and an agreement form. Prompt return of this form will allow us to use your program as soon as possible.

18) If you want the program returning to you, should we find it unsuitable for publication, then you should enclose a stamped self addressed envelope.

19) The last and most important point to make is 'get writing', we are waiting for your articles.

Figure 1



Electronic Notebook

Get yourself organised with this powerful database

By L. Keighley

Electronic Notebook is an electronic implementation of the very popular Personal Organiser. It provides the means by which information can be stored, under a key, in a file. It does this by utilising database techniques similar to those on mainframe computers.

An Electronic Notebook file can be as small or as large as you require, up to the maximum size of the disk. On the 1541 disk drive this will be around 3500 records, but on larger disks with a capacity of 1 megabyte the number increases to 24000. A single Electronic Notebook file contains four different record types which are used to implement the database structure (See Figure 1). The records form a hierarchical structure at which each level holds varying degrees of information required by the user. The four records are:

CONTROL This record holds control information about the usage of the database. This includes details of where the first empty record is and what the access password is. The user is not given access to this record it is maintained by the Electronic Notebook program alone.

SECTION This record is the Header or Folder under which information is to be stored. Apart from pointers to the next and last Section records and pointers to its next and last Entry Key records, it holds the Title of the section and its type. The types allowed are Labels, Notes and Text.

ENTRY KEY This record holds the key to the information that is to be stored below it, along with the pointer to the first Entry record. Pointers are also held to the next and last Entry Key records.

ENTRY Each Entry record holds up to 35 characters of information along with the pointer to the next Entry. If the information to be held is of the Text type a seventy character line is made up of two consecutive Entry records.

The Section records can only be accessed by stepping through all those present, but the Entry Key records can be obtained using the given Key.

All the records at each lever are 'owned' by a record at the level above. For example, a Section record may be designated as 'Addresses' and may have below it two address Entry Keys for 'Fred' and 'Joan' below each of the Entry Key records will be any number of Entry records that it takes to hold all of the address for each person. The records held at all levels below Control are held in 'chains'.

The program allows maintenance of all user information, add/delete/insert, at the lowest three levels with the exception of key information, this is restricted by password. All the 'chains' are held in sequence of key and to change the key requires that their position in the chain is altered. While the physical position on the disk of the record does not change all the pointers around its old and new positions must be amended. This action requires the reading and writing of a number of records and therefore takes some time.

The database could be used as a diary if each Entry Key within a Section called 'Diary' was given the date of each day in the year.

The hardcopy output facilities provided by Electronic Notebook are Labels, printed in single or two up form, Notes, again printed in single or two up form and Text printed as seventy character lines.

Access to the facilities of Electronic Notebook are from an on-screen menu. Functions are selected by keying the appropriate number. Some require two numbers to be keyed, the first to select general function and the second to select the specific function.

Control:
E First available empty record
E1 End of file + 1 record
S1 First Section record
S2 Last Section record
M Maximum number of records
M1 Marker Types used
P$ Password

Section:
T$ Title
S3 Next Section
S4 Last Section
E2 Next Entry
E3 Last Entry
T1$ Type
M2 Marker

Entry Key:
E4 Next Entry
E5 Last Entry
K$ Key
N1 Next Entry

Entry:
N2 Next Entry
D$ Data

Operating Instructions

1 Load and Run the program 'ENOTEBOOK', a menu will then be displayed of all the actions that are allowed on the database.

2 To select an action, with the exception of 'Initialise', requires that *two* keys be pressed. The first selects the area of the database that is to be worked on or the print option and second selects the type of action.

3 When entering Text information the ',' cannot be used. This is because of the way in which the computer takes data in from the keyboard. The comma is used to indicate the end of input for one field and the beginning of input for the next. So that, the comma can be entered in text the '] ' should be used. This will result in a comma being displayed on the screen and printed where necessary.

4 '9' must always be selected from the menu to end the program this ensures that the database is closed correctly and the last buffer of information is stored on the disk.

5 When a new database is set up the number of database records allowed can be varied from 500 up to 3500. Each record has to be written to the database and therefore this takes some time. The database already set up has 2500 entries and took approximately 10 minutes to intialise.

6 The password when correctly entered allows the pointer fields to be amended. This can corrupt the database if done incorrectly, hence the password. The password for the sample database is 'CDUCDU'.

Head for Home

Challenge your friends to a version of one of the World's oldest boardgames

Head for Home is presented in the form of two Basic Loaders. These should be typed in and saved seperately. If using tape make sure that you alter the DLOAD at the end of the first program to a LOAD instruction.

Care should be taken when entering the program, as some of the lines contain machine code. A single mistake on entry would cause the program not to work.

Getting Going

Once the program is up and running, you will be asked if you require instructions and the number of players. Head for Home is designed for 2 – 6 players.

To start the game, a player needs to throw a six on the computer dice. The play rotates around the players until one scores a six. If no move is possible a sound will tell you. Once a player throws a six the dice is rolled again until a different number is thrown. The sum of the dice throws becomes the number of positions that a player can move, though it costs a six to move a man onto the board.

When a move is required a cursor will appear on the screen. The Player should press one of the numbers 1 – 4, depending on which piece they wish to move. The move is only accepted if it is legal. If no move can be made, press P to pass.

Pieces move in the same order as the die throws. So, if you have two sixes and a three and press 2 4 1, pices 2 and 4 move 6 spaces and piece 1 moves 3. Entering 2 2 2 would move piece 2 15 spaces.

Each playing piece travels around the board once. When it leaves the white track and moves onto the last 5 coloured spaces. The exact number must be thrown in order to reach home base.

That's all there is to it – have fun!

Software for Sale

If you think that one of our programs looks very interesting, but you can't afford the time to type it in, then our software service will help you out

It's three o'clock in the morning. You sit at the computer keyboard having just finished a marathon typing session entering one of the superb programs from *Your Commodore*. Your fingers reach for the keyboard and press the letters R, U and N. You press RETURN, sit back and nothing happens.

Everyone has probably faced this problem. When it does happen it's a matter of spending hours searching through the program for any typing mistakes. No matter how long you look or how many people help you, you can usually guarantee that at least one little bug slips through unnoticed.

The *Your Commodore* Software Service makes available all of the programs from each issue on both cassette and disk at a price of £6.00 for disk and £4.00 for cassette. None of the documentation for the programs is supplied with the software since it is all available in the relevant magazine. Should you not have the magazine then back issues are available from the following address:

INFONET LTD, 5 River Park Estate, Berkhamsted, Herts HP4 1HL.
Tel: (04427) 76661

Please contact this address for prices and availability.

The Disk

Programs on the disk will also be supplied as totally working versions, i.e. when possible we will not use Basic Loaders thus making use of the programs much easier. Unfortunately at the moment we cannot duplicate C16 and Plus/4 cassettes. However programs for these machines will be available on the disk.

What programs are available?

At the top of each article you will find a strap containing the article type, C64 Program etc. So that you can see which programs are available on which format, you will also find a couple of symbols after this strap. The symbols have the following meaning:

This symbol means that the program is available on cassette.

These programs are available on disk.

Please Note

Since the programs supplied on cassette are total working versions of the program, we do not put disk-only programs on tape. There is no sense in placing a program that expects to be reading from disk on to tape.

OCTOBER 1988

SPRITE LIBRARY – In this instalment our sprites take on the look of the alphabet (C64)
SAMPLER 64 – See September 1988 for details (full program on both September and October disks and tapes).
SET THE ALARM – Use the C64's in-built clock as an alarm.
JACK IN THE BOX – A handy box utility for Basic and machine code programmers (C64).
FILE EXTENSION – Modify your disk directory so that you can load all your programs with ease (C64 disk only).
HI-RES/MULTI/COLOUR PLOTTER – A selection of plot routines for both Basic and machine code programmers (C64).
JOYSTICK CURSOR – Let your joystick emulate the cursor keys (C64).
HI-RES FILL – A superb machine code fill routine for all graphics programmers (C64).
CODE RELOCATION – Change your machine code programs so that they will run in any area of your C64's memory.

ORDER CODE
TAPE YCOCT88 £4.00
DISK YDOCT88 £6.00

NOVEMBER 1988

SECRET WRITING — Learn how to conceal messages and how to protect your Basic programs (C64).
PAINLESS WINDOWS — Extend the power of the C128's WINDOW command.
UNLOCK HIDDEN 128 POWERS — Use the full keyset of your C128 in C64 mode.
WILLIAM TELL — Fight off the Austrian soldiers in this great C64 arcade game.
TALKING COMPUTERS — Use speech synthesisers with your C64.

ORDER CODE
TAPE YCNOV88 £4.00
DISK YDNOV88 £6.00

DECEMBER 1988

F-KEY LABELLER – Use your Commodore printer to produce a template for your function keys (C64).
+4 DIRECTORY EDITOR – Rearrange your Plus/4 disk directory entries however you like. Available on disk only.
DOUBLE HEIGHT – Enhance your C64 text displays with a double height character set.
PROGRAM PROTECTION – Protect your latest C64 masterpiece from prying eyes.
SPRITE CONTROLLER – Basic sprite manipulation made easy. C64.
GRAPHIC EDITOR – A quick and simple C64 UDG editor.
MASTERING THE RASTER – The source code to accompany our article on raster interrupts (C64).
CREATING CHARACTERS – Give your Plus/4 and Commodore printer a better print quality. Available on disk only.
WILLIAM TELL – See November for details (C64).

ORDER CODE
TAPE YCDEC88 £4.00
DISK YDDEC88 £6.00

JANUARY 1989

PREFAB SPRITES – A powerful sprite editor for the C64.
DAZPRO – A simple but helpful text processor for the C64. Available on disk and cassette but will only store files on tape.
UDG COMPRESSOR – Save on memory when using UDG's in your programs. For C64 only.
WILLIAM TELL – Our popular arcade game for the C64.
+4 AUTORUN – Improve tape loading on your Plus/4 cassette. Only available on disk.
MINIBASE – A database for C128 owners.

ORDER CODE
DISK YDJAN89 £6.00
TAPE YCJAN89 £4.00

FEBRUARY 1989

TAPE MENU – Add a menu system to your program cassettes (C64).
SONIC EFFECTS – A superb sound editor for the C64.
F DUMP – Dump your C64 text screens to printer with ease.
DATA LOADER – A simple way to enter those reams of C64 DATA lines.
SPRITE LIBRARY – A collection of birds to your growing library (C64).
PLAY THE GAME – A superb fruit machine programe for the Plus/4. (Available on disk only).

ORDER CODE
DISK YDFEB89 £6.00
TAPE YCFEB89 £4.00

Cassettes or disks are available from March 1986. Please ring the editorial office (01-437 0626) for details of these.

ORDER FORM – PLEASE COMPLETE IN BLOCK CAPITALS

NAME	QTY	TAPE/DISK	ORDER CODE	PRICE
MARCH 89		**TAPE (£4.00)**	**YCMAR89**	
MARCH 89		**DISK (£6.00)**	**YDMAR89**	
OVERSEAS POST £1				
			TOTAL	

NAME...

ADDRESS...

POSTCODE...

I enclose a cheque/postal order for £...... made payable to ARGUS SPECIALIST PUBLICATIONS LTD.

All orders should be sent to: YOUR COMMODORE, READERS SERVICES, ARGUS SPECIALIST PUBLICATIONS, 9 HALL ROAD, HEMEL HEMPSTEAD, HERTS HP2 7BH.
Please allow 28 days for delivery.

Listings

Get it right first time with our deluxe program system for the C64.

You may have noticed that our listings are free of those horrible little black blobs which send you searching around the keyboard for a suitable graphic symbol. You may also have noticed the funny numbers by the side of each line of the listing. Fret no more, it's all part of our easy entry aid.

Instead of those nasty graphics and rows of countless spaces in PRINT statements and strings we use a special coding system. The code, or mnemonic, is always contained in square brackets and you'll soon learn to decipher their meanings.

For example, [SA] would mean type in a Shifted A, or an ace of spades in layman's terms, and [SA10] would mean a row of ten of these symbols.

[S+2] means hold down the shift key and press the plus key twice. It doesn't take a great leap of logic to realise that [C+2] means exactly the same thing except that the Commodore key (bottom left of the keyboard) is held down instead of the shift key.

If more than two spaces appear in a statement then this will be printed as [SPC4] or, exceptionally, [SSPC4]. Translated into English this means press the spacebar four times or in the latter case hold the shift key down while you do it.

A string of special characters could appear as:
[CTRL N, DOWN2,LEFT5,BLUE, F3,C3]
This would be achieved by holding down the CTRL key as you press N, press the cursor key down twice, the cursor left key five times, press the key marked BLUE while holding down the CTRL key, press the F3 key and, finally hold the Commodore key down while pressing the number two key (C2 would of course make the computer print in brown).

Always remember that you should only have a row of graphics characters on your screen with no square brackets and no commas, unless something like this appears:
[SS],[C*]
In this case the two characters should have a comma between them.

On rare occasions [REV T] will appear in a listing. This is a delete symbol and is created by entering the line up to this mnemonic. Then type a closing quotation mark (SHIFT & 2) and delete it. This gets the computer out of quotes mode. Hold down CTRL and press the number nine key (RVSON), type the relevant number of reversed T's and then hold down CTRL and press zero (RVSOFF). Next type another quotation mark and delete it again. Now finish the line and press RETURN.

A list of these special cases is given in the table but remember that only one of these mnemonics will appear outside of a PRINT string: the symbol for pi. This may appear when its value is needed in a calculation so this may look something like:
:CC=2*[PI]*R:
Ignore the square brackets and just type in a shifted upward pointing arrow (ie. the pi symbol).

PROGRAM: SYNTAX CHECKER

```
5 REM SYTAX CHECKER - ERIC DOYLE

10 BL=10    :LN=70    :SA=49152
20 FOR L=0 TO BL:CX=0:FOR D=0 TO
 15
30 READ A:IF A>255THENPRINT"NUMB
ER TO LARGE";LN+(L*10):STOP
40 CX=CX+A:POKE SA+L*16+D,A:NEXT
 D
50 READ A:IF A><CX THENPRINT"ERR
OR IN LINE";LN+(L*10):STOP
60 NEXT L:SYS 49152:NEW
70 DATA 173,5,3,201,165,208,31,1
20,169,9,141,32,208,141,33,208,1
847
80 DATA 169,7,141,134,2,169,13,3
2,210,255,169,64,141,4,3,169,168
2
90 DATA 192,141,5,3,88,96,120,16
9,124,141,4,3,169,165,141,5,1566

100 DATA 3,169,14,141,134,2,141,
32,208,169,6,141,33,208,88,96,15
85
110 DATA 32,124,165,72,138,72,15
2,72,162,0,165,20,133,254,165,21
,1747
120 DATA 24,101,254,133,254,189,
0,2,240,18,69,254,133,254,232,18
9,2346
130 DATA 0,2,240,8,24,101,254,13
3,254,232,208,233,169,1,141,134,
2134
140 DATA 2,165,254,74,74,74,74,3
2,156,192,32,210,255,165,254,41,
2054
150 DATA 15,32,156,192,32,210,25
5,169,13,32,210,255,169,13,32,21
0,1995
160 DATA 255,169,7,141,134,2,104
,168,104,170,104,96,24,105,48,20
1,1832
170 DATA 58,16,1,96,24,105,7,96,
0,0,0,0,0,0,0,0,403
```

by Eric Doyle

Checksum Program

The hexadecimal numbers appearing in a column to the left of the listing should not be typed in with the program. These are merely checksum values and are there to help you get each line right. Don't worry if you don't understand the hexadecimal system, as long as you can compare two characters on the screen with the corresponding two characters in the magazine you can use our line checking program.

Type in the Checksum Program, make sure that you've not made any mistakes and save it to tape or disk immediately because it will be used with most of the present and future listings appearing in Your Commodore.

At the start of each programming session, load Checksum and run it. The screen will turn brown with yellow characters and each time you type in a line and press the RETURN key a number will appear on the screen in white. This should be the same as the corresponding value in the magazine.

If the two values don't relate to one another, you have not copied the line exactly as printed so go back and check each character carefully. When you find the error simply correct it and press RETURN again.

If you want to turn off the checker simply type SYS49152 and the screen will return to the familiar blue colours. You can then do whatever it was you wanted to do and if this doesn't use the area where Checksum lies you can go back to it with the same SYS command.

No system is foolproof but the chances of two errors cancelling one Many of the listings are presented in lower case. To turn your computer to lower case mode press the Commodore key and the SHIFT key at the same time.

Mnemonic	Symbol	Keypress
[RIGHT]		CRSR left/right
[LEFT]		SHIFT & CRSR left/right
[DOWN]		CRSR up/down
[UP]		SHIFT & CRSR up/down
[F1]		f1 key
[F2]		SHIFT & f1 key
[F3]		f3 key
[F4]		SHIFT & f3 key
[F5]		f5 key
[F6]		SHIFT & f5 key
[F7]		f7 key
[F8]		SHIFT & f7 key
[HOME]		CLR/HOME
[CLR]		SHIFT & CLR/HOME
[RVSON]		CTRL & 9
[RVSOFF]		CTRL & 0

Mnemonic	Symbol	Keypress
[BLACK]		CTRL & 1
[WHITE]		CTRL & 2
[RED]		CTRL & 3
[CYAN]		CTRL & 4
[PURPLE]		CTRL & 5
[GREEN]		CTRL & 6
[BLUE]		CTRL & 7
[YELLOW]		CTRL & 8
[POUND]		£
[LARROW]		←
[UPARROW]		↑
[PI]		SHIFT & ↑
[INST]		SHIFT & INST/DEL
[REV T]		see text
[Cletter]		CBM + letter
[Sletter]		SHIFT + letter

LIFESAVER	C64	REVERSER	1/1

This little machine code routine will enable Basic programs to be enhanced with flashing messages etc. It will reverse on and off any part of the screen that you desire.

The routine redirects the IRQ Interrupt Vector at memory location 788 so your Basic program will be free do do what you want.

To start REVERSER, POKE the data into memory then POKE the following locations with the values requested:
POKE 251,L-BYTE (Screen start address)
POKE 252,HI-BYTE
POKE 253,NUMBER OF CHARS
POKE 254,FLASH RATE
Then SYS 49152 to start or
SYS 49196 to switch off.

The program includes a short demo to show how to set the program up.

```
10 REM **** REVERSER MACHINE CODE ****
20 FORL=49152TO49215:READA:POKEL,A:D=D+A:NEXT
30 IFD<>8033THENPRINT"DATA ERROR":STOP
40 DATA 120, 169, 13, 141, 20, 3 ,169 ,192, 141, 21, 3, 88,96,
173, 57, 192
50 DATA 197, 254, 208, 18, 169, 255, 141, 57, 192, 160, 0, 177,
251, 73, 128, 145
60 DATA 251, 200, 196, 253, 208, 245, 238, 57, 192, 76, 49, 234,
120,169, 49, 141
70 DATA 20, 3, 169, 234, 141, 21, 3, 88, 96, 0, 31, 168, 255, 8,
21, 74
80 REM **** REVERSER DEMO ****
90 POKE53280,0:POKE53281,0:POKE 646,7
100 PRINT CHR$(147);:PRINT TAB(13)"YOUR
COMMODORE"
110 POKE253,18:REM * NUMBER OF CHARACTERS TO
REVERSE *
120 POKE254,25: REM * FLASH RATE *
130 A=1035: REM * SCREEN ADDRESS *
140 POKE251,A AND 255: REM * POKE VALUES FOR
SCREEN ADDRESS *
150 POKE252,A/256
160 SYS 49152: REM * CALL M/C *
170 GET A$:IFA$=""THEN 170
180 SYS 49196: REM * SWITCH OFF M/C *
```

By Neil Higgins

Listings

William Tell

PROGRAM: WILLIAM TELL - LISTING 5

```
7F   10 BL=425   :LN=50    :SA=3123
     2
F9   20 FOR L=0 TO BL:CX=0:FOR D=
     0 TO 15
8B   21 READ A:POKE53280,A
9E   22 CX=CX+A:POKE SA+L*16+D,A:
     NEXT
A2   30 READA:IF A=CX THEN40
12   31 PRINT"ERROR IN LINE";LN+(
     L*10):STOP
1B   40 NEXT L:SYS37872
44   50 DATA 32,31,122,32,7,122,9
     6,120,169,182,141,20,3,169,1
     22,141,1509
0D   60 DATA 21,3,169,0,141,188,2
     ,141,18,208,32,157,122,88,96
     ,169,1555
AE   70 DATA 14,141,134,2,169,147
     ,32,210,255,173,22,208,41,24
     7,9,16,1820
35   80 DATA 141,22,208,162,7,142
     ,210,2,169,0,141,32,208,141,
     97,3,1685
92   90 DATA 141,98,3,133,251,133
     ,252,157,56,3,157,64,3,157,7
     2,3,1683
4D   100 DATA 157,80,3,202,16,241
     ,169,3,141,229,2,164,251,185
     ,71,126,2040
ED   110 DATA 240,38,168,166,252,
     189,89,126,170,230,251,230,2
     52,24,32,240,2697
3C   120 DATA 255,169,35,32,210,2
     55,169,36,32,210,255,169,37,
     32,210,255,2361
DF   130 DATA 169,38,32,210,255,7
     6,91,122,169,210,141,49,3,16
     9,200,141,2075
3F   140 DATA 48,3,169,169,141,50
     ,3,169,100,141,96,3,96,173,1
     7,208,1586
03   150 DATA 41,127,141,17,208,1
     73,26,208,9,1,141,26,208,173
     ,14,220,1733
01   160 DATA 41,254,141,14,220,9
     6,173,25,208,141,25,208,41,1
     ,208,3,1799
CA   170 DATA 76,49,234,172,188,2
     ,240,3,76,103,123,141,27,208
     ,169,254,2065
47   180 DATA 141,28,208,136,140,
     23,208,140,29,208,173,59,3,2
     40,4,206,1946
32   190 DATA 27,208,136,140,21,2
     08,169,0,141,33,208,169,104,
     141,18,208,1931
45   200 DATA 169,3,141,38,208,16
     9,6,141,37,208,160,7,162,14,
     185,19,1667
43   210 DATA 126,153,248,7,169,1
     ,153,39,208,185,64,3,217,11,
     126,176,1886
9D   220 DATA 5,105,2,153,64,3,15
     7,0,208,169,60,157,1,208,202
     ,202,1696
11   230 DATA 136,208,219,173,19,
     126,141,248,7,169,0,141,39,2
     08,173,58,2065
34   240 DATA 3,208,11,24,173,64,
     3,105,2,141,64,3,144,27,169,
     1,1142
```

```
27   250 DATA 141,58,3,141,16,208
     ,24,173,64,3,105,2,141,64,3,
     141,1287
03   260 DATA 0,208,201,50,144,3,
     141,59,3,141,0,208,169,60,14
     1,1,1529
F6   270 DATA 208,238,188,2,76,12
     9,234,192,1,208,105,173,59,3
     ,240,89,2145
12   280 DATA 169,63,141,21,208,1
     41,23,208,141,28,208,141,29,
     208,169,0,1898
2D   290 DATA 141,16,208,169,12,1
     41,38,208,169,2,141,37,208,1
     60,5,162,1817
DB   300 DATA 10,185,65,126,153,2
     48,7,169,7,153,39,208,185,72
     ,3,217,1847
B9   310 DATA 59,126,176,5,105,2,
     153,72,3,157,0,208,169,112,1
     57,1,1505
E2   320 DATA 208,202,202,136,16,
     219,169,5,141,39,208,141,44,
     208,173,0,2111
49   330 DATA 208,205,59,126,144,
     3,141,60,3,238,188,2,169,154
     ,141,18,1859
54   340 DATA 208,76,129,234,192,
     2,208,116,173,60,3,240,100,1
     62,10,142,2055
5D   350 DATA 38,208,202,142,37,2
     08,162,0,142,16,208,232,142,
     23,208,142,2110
B1   360 DATA 27,208,169,193,141,
     29,208,169,255,141,21,208,17
     3,61,3,240,2246
1C   370 DATA 3,206,21,208,169,6,
     141,28,208,162,14,160,7,185,
     51,126,1695
61   380 DATA 153,248,7,185,43,12
     6,153,39,208,185,80,3,217,27
     ,126,176,1976
E5   390 DATA 5,105,2,153,80,3,15
     7,0,208,185,35,126,157,1,208
     ,202,1627
42   400 DATA 202,136,16,217,173,
     0,208,201,254,144,6,141,61,3
     ,206,27,1995
24   410 DATA 208.238.188.2.169.2
     02,141,18,208,76,129,234,169
     ,0,141,18,2141
9B   420 DATA 208,141,188,2;173,6
     1,3,240,114,169,1,141,33,208
     ,169,208,2059
CD   430 DATA 141,1,208,162,0,142
     ,39,208,142,29,208,232,172,6
     3,3,208,1958
C1   440 DATA 17,142,28,208,202,1
     42,37,208,169,10,141,38,208,
     169,5,141,1865
59   450 DATA 39,208,173,62,3,240
     ,12,173,98,3,208,7,32,126,12
     5,169,1678
74   460 DATA 3,208,2,169,1,141,2
     1,208,141,23,208,185,48,3,20
     6,210,1777
C2   470 DATA 2,208,8,169,4,141,2
     10,2,32,37,125,141,248,7,173
     ,57,1564
96   480 DATA 3,208,76,24,173,56,
     3,105,1,144,10,72,169,1,141,
     16,1202
7F   490 DATA 208,141,57,3,104,14
     1,56,3,141,0,208,173,22,208,
     41,7,1513
B5   500 DATA 208,39,32,202,125,1
     73,61,3,240,21,238,49,126,23
     8,50,126,1931
21   510 DATA 173,49,126,41,15,24
     0,243,173,52,126,73,24,141,5
     2,126,173,1827
96   520 DATA 22,208,9,7,141,22,2
     08,208,3,206,22,208,76,49,23
     4,169,1792
1F   530 DATA 1,141,16,208,24,169
     ,1,109,56,3,201,86,144,183,2
     00,192,1734
D7   540 DATA 3,144,2,160,0,140,6
     3,3,206,229,2,169,0,141,56,3
     ,1321
EB   550 DATA 141,57,3,240,166,19
     2,0,208,16,24,173,48,3,105,1
     ,201,1578
08   560 DATA 204,144,2,169,200,1
     53,48,3;96,192,1,208,14,24,1
     73,49,1680
BF   570 DATA 3,105,1,201,213,144
     ,238,169,209,208,234,56,173,
     62,3,240,2259
B5   580 DATA 15,173,50,3,233,1,2
     01,162,176,2,169,165,141,50,
     3,96,1640
1A   590 DATA 173,50,3,233,1,201,
     166,176,2,169,169,72,173,98,
     3,208,1897
BF   600 DATA 10,173,56,3,201,100
     ,144,3,141,62,3,104,208,222,
     169,0,1599
7A   610 DATA 141,28,208,141,40,2
     08,169,178,141,249,7,169,208
     ,141,3,208,2239
5C   620 DATA 24,173,97,3,240,28,
     169,2,141,16,208,173,96,3,10
     5,4,1482
38   630 DATA 201,86,144,31,141,9
     8,3,169,0,141,62,3,169,169,1
     41,50,1608
6F   640 DATA 3,96,173,96,3,105,4
     ,144,10,72,169,2,141,16,208,
     141,1383
EA   650 DATA 97,3,104,141,96,3,1
     41,2,208,96,169,0,133,253,16
     9,4,1619
11   660 DATA 133,254,169,255,133
     ,251,169,3,133,252,162,10,16
     0,0,177,253,2514
8E   670 DATA 72,200,177,253,145,
     251,200,192,40,208,247,104,1
     45,251,202,48,2735
14   680 DATA 25,24,169,80,101,25
     3,133,253,165,254,105,0,133,
     254,169,80,2198
4C   690 DATA 101,251,133,251,144
     ,214,230,252,208,210,96,254,
     73,109,131,159,2816
BC   700 DATA 187,214,254,191,179
     ,180,181,181,180,182,183,254
     ,230,110,150,174,3030
CE   710 DATA 198,136,184,166,166
     ,166,166,166,166,180,180,0,5
     ,7,3,3,1892
18   720 DATA 3,1,1,191,222,187,1
     88,189,190,175,176,252,208,1
     80,152,120,2435
6B   730 DATA 74,186,181,181,185,
     184,186,1,9,15,19,20,13,25,5
     ,35,1319
56   740 DATA 30,26,2,28,36,15,21
     ,31,0,0,2,4,6,8,10,4,223
0B   750 DATA 12,18,14,0,6,8,10,1
     2,18,16,234,234,234,234,234,
     234,1518
1F   760 DATA 0,255,0,255,0,255,0
     ,255,0,255,0,255,0,255,0,255
     ,2040
AF   770 DATA 65,66,66,65,66,65,6
     5,66,66,65,66,65,66,66,65,66
     ,1049
D1   780 DATA 65,66,66,66,65,66,6
     3,63,63,63,66,65,66,66,66,66
     ,1041
09   790 DATA 65,66,65,63,63,63,6
     6,66,65,66,65,211,211,211,21
     1,211,1768
9D   800 DATA 211,211,211,211,211
     ,211,211,211,211,211,211,65,
     66,65,63,63,2643
87   810 DATA 63,63,66,66,73,65,7
     3,65,73,65,66,63,63,63,66,65
     ,1058
43   820 DATA 66,65,66,211,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,792
0    830 DATA 32,32,211,65,65,66,
     63,63,63,63,66,65,74,65,74,6
     5,1132
I9   840 DATA 74,65,66,63,63,63;6
     6,66,65,66,65,211,211,211,21
     1,211,1777
:3   850 DATA 211,211,211,211,211
     ,211,211,211,211,211,211,65,
     66,65,63,63,2643
I8   860 DATA 63,63,66,65,66,66,6
     5,66,65,66,65,63,63,63,64,65
     ,1034
'5   870 DATA 65,65,66,211,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,791
29   880 DATA 32,32,211.65,210,65
     ,63,63,63,63,66,66,65,66,66,
     65,1261
IB   890 DATA 66,66,65,63,63,63,6
     4,66,65,66,65,211,32,32,32,3
     2,1051
L8   900 DATA 32,32,32,32,32,32,3
     2,32,32,32,211,65,66,65,63,6
     3,853
21   910 DATA 63,63,66,65,73,65,7
     3,65,73,210,65,63,63,63,66,6
     5,1201
2B   920 DATA 65,65,66,211,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,791
74   930 DATA 32,32,211,65,210,65
     ,63,63,63,63,64,65,74,65,74,
     65,1274
06   940 DATA 74,65,65,63,63,63,6
     4,66,65,66,66,211,32,32,32,3
     2,1059
5E   950 DATA 32,32,32,32,32,32,3
     2,32,32,32,211,65,66,65,63,6
     3,853
32   960 DATA 63,63,64,65,66,65,6
     6,65,65,66,65,63,63,63,64,65
     ,1031
DB   970 DATA 65,66,65,211,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,791
0E   980 DATA 32,32,211,65,210,65
     ,63,63,63,63,64,66,65,66,65,
     66,1259
25   990 DATA 66,210,65,63,63,63,
     64,66,65,65,66,211,32,32,32,
     32,1195
04   1000 DATA 32,32,32,32,32,32,
     32,32,32,32,211,65,66,65,63,
     63,853
13   1010 DATA 63,63,64,65,73,65,
     73,65,73,65,65,63,63,63,66,6
     5,1054
A5   1020 DATA 66,65,55,211,32,32
     ,32,32,32,32,32,32,32,32,32,
     32,791
63   1030 DATA 32,32,211,65,210,6
     5,63,63,63,63,64,66,74,65,74
     ,65,1275
9D   1040 DATA 74,210,65,63,63,63
     ,64,65,65,66,66,211,32,32,32
     ,32,1203
0A   1050 DATA 32,32,32,32,32,32,
     32,32,32,32,211,65,66,65,63,
     222,1012
B6   1060 DATA 223,166,64,65,66,6
     6,65,66,65,66,65,63,63,63,66
     ,66,1298
EC   1070 DATA 66,66,65,211,32,32
     ,32,32,32,32,32,32,32,32,32,
     32,792
B4   1080 DATA 32,32,211,65,210,6
     5,63,221,100,166,64,66,128,6
     5,65,128,1681
B4   1090 DATA 65,210,65,63,63,63
```

```
    ,64,65,65,65,66,211,32,32,32
    ,32,1193
72  1100 DATA 32,32,32,32,32,32,
    32,32,32,32,211,65,66,65,63,
    221,1011
01  1110 DATA 100,166,64,165,127
    ,65,66,127,65,66,65,63,63,63
    ,66,65,1396
35  1120 DATA 66,66,65,211,211,2
    11,211,211,211,211,211,211,2
    11,211,211,211,2940
64  1130 DATA 211,211,211,65,210
    ,65,63,221,100,166,64,165,12
    7,65,66,127,2137
FF  1140 DATA 65,66,65,63,63,63,
    64,66,68,68,68,68,68,68,68,6
    8,1059
6E  1150 DATA 68,68,68,68,68,68,
    68,68,68,68,68,68,68,68,67,2
    21,1240
6D  1160 DATA 100,166,64,66,66,6
    6,65,66,66,210,65,63,63,63,6
    4,65,1318
CB  1170 DATA 39,32,40,39,39,40,
    32,39,40;32,40,39,39,40,40,3
    9,609
87  1180 DATA 39,40,39,40,39,40,
    39,212,220,166,64,65,66,225,
    226,65,1585
8F  1190 DATA 66,66,65,63,63,63,
    64,65,39,32,40,39,39,40,39,3
    9,822
23  1200 DATA 39,39,40,39,40,39,
    32,39,39,32,40,39,39,40,39,4
    0,615
BA  1210 DATA 212,63,64,66,66,22
    7,230,66,65,66,65,63,63,63,6
    6,66,1511
DC  1220 DATA 40,39,40,32,39,32,
    40,39,39,32,32,39,32,32,39,3
    2,578
39  1230 DATA 32,40,39,32,39,40,
    40,39,32,67,66,65,66,66,65,6
    6,794
7F  1240 DATA 66,210,65,63,63,63
    ,64,65,32,32,39,32,32,40,32,
    32,930
A0  1250 DATA 32,40,39,40,32,39,
    32,39,32,32,39,32,32,39,32,3
    2,563
E9  1260 DATA 32,39,32,39,32,39,
    39,32,39,32,39,224,63,63,64,
    66,874
1A  1270 DATA 40,32,39,32,40,32,
    39,32,40,32,32,32,39,32,32,3
    2,557
54  1280 DATA 39,32,32,39,32,39,
    32,32,39,32,32,32,39,32,39,3
    2,554
84  1290 DATA 32,32,39,32,224,63
    ,66,65,6,15,18,9,61,48,20,15
    ,745
2C  1300 DATA 57,48,48,58,16,79,
    52,57,49,53,50,43,9,44,16,69
    ,748
3E  1310 DATA 40,49,48,50,52,43,
    9,41,58,14,69,32,32,32,32,32
    ,633
CE  1320 DATA 32,32,32,32,32,32,
    32,32,32,32,32,32,32,32,32,3
    2,512
02  1330 DATA 32,32,32,32,0,255,
    0,255,0,255,0,255,0,255,0,25
    5,1658
8F  1340 DATA 97,97,97,97,97,97,
    97,97,97,97,97,97,97,97,97,9
    7,1552
0D  1350 DATA 97,97,97,97,97,97,
    97,97,97,97,97,97,97,97,97,9
    7,1552
DB  1360 DATA 97,97,97,97,97,97,
    97,97,97,97,97,97,97,97,97,9
    7,1552
C9  1370 DATA 97,97,97,97,97,97,
    97,97,97,97,97,97,97,97,97,9
    7,1552
0A  1380 DATA 97,97,97,97,97,97,
    97,97,94,95,97,97,97,97,97,9
    7,1547
CC  1390 DATA 158,97,94,95,97,97
    ,97,97,94,96,95,97,97,97,97,
    97,1602
1A  1400 DATA 97,97,97,97,97,97,
    97,97,94,95,97,97,97,97,97,9
    0,1540
0C  1410 DATA 100,92,91,97,94,95
    ,97,97,161,90,100,100,91,96,
    95,90,1586
BF  1420 DATA 93,100,100,91,97,9
    7,94,96,95,97,97,94,95,97,97
    ,90,1530
11  1430 DATA 100,100,91,97,97,9
    7,90,93,100,100,100,92,100,9
    2,91,97,1537
88  1440 DATA 156,100,100,92,100
    ,100,93,100,100,100,92,100,9
    1,90,100,93,1607
65  1450 DATA 100,91,90,100,92,9
    1,90,93,100,92,100,91,97,90,
    93,100,1510
0D  1460 DATA 100,100,100,100,92
    ,100,100,91,115,156,100,100,
    92,93,100,47,1586
C7  1470 DATA 100,100,100,100,10
    0,47,100,100,92,100,100,100,
    100,93,93,100,1525
54  1480 DATA 93,100,100,100,93,
    100,100,100,100,100,100,100,
    86,85,86,85,1528
48  1490 DATA 115,115,156,100,93
    ,100,100,27,43,44,44,44,44,2
    9,43,100,1197
83  1500 DATA 100,100,100,100,10
    0,100,100,100,100,100,100,93
    ,100,100,100,100,1593
5D  1510 DATA 100,100,100,100,88
    ,82,83,81,115,115,115,113,11
    2,112,41,28,1485
54  1520 DATA 42,46,46,30,46,46,
    42,122,41,114,112,122,41,122
    ,41,122,1135
0A  1530 DATA 122,41,122,41,41,1
    22,41,41,122,122,41,41,63,63
    ,84,65,1172
8A  1540 DATA 117,117,118,114,32
    ,32,39,122,41,41,41,33,41,41
    ,41,40,1010
01  1550 DATA 32,32,32,32,32,32,
    32,32,32,32,32,32,32,32,32,3
    2,512
4E  1560 DATA 32,32,32,32,63,63,
    64,73,109,109,73,32,32,32,32
    ,32,842
9D  1570 DATA 32,32,32,32,32,32,
    32,32,32,32,32,32,32,32,32,3
    2,512
76  1580 DATA 32,32,32,32,32,32,
    32,32,32,32,32,32,63,63,64,7
    4,648
CE  1590 DATA 109,109,74,60,61,6
    0,61,60,61,60,61,60,61,60,61
    ,60,1078
32  1600 DATA 61,60,61,60,61,60,
    61,60,61,60,61,60,61,60,61,6
    0,968
CB  1610 DATA 61,60,61,60,103,63
    ,64,65,109,109,104,66,65,66,
    66,65,1187
2F  1620 DATA 66,65,66,66,65,66,
    66,65,66,66,65,66,65,66,66,6
    5,1050
65  1630 DATA 66,66,66,65,66,65,
    66,66,65,66,65,66,209,63,64,
    123,1247
F7  1640 DATA 70,69,104,66,66,65
    ,66,65,1,2,3,4,5,6,7,8,607
B7  1650 DATA 9,10,11,12,13,14,1
    5,16,17,18,19,20,21,22,23,24
    ,264
45  1660 DATA 25,26,66,210,209,6
    3,64,125,72,71,104,79,79,79,
    79,79,1430
FA  1670 DATA 79,79,79,79,79,79,
    79,79,79,79,79,79,79,79,79,7
    9,1264
DD  1680 DATA 79,79,79,79,79,79,
    79,79,79,79,79,79,79,103,64,
    65,1259
D3  1690 DATA 106,106,107,80,80,
    80,80,80,80,80,80,80,80,80,8
    0,80,1359
82  1700 DATA 80,80,80,80,80,80,
    80,80,80,80,80,80,80,80,80,8
    0,1280
E9  1710 DATA 80,80,80,80,80,80,
    80,80,109,109,104,66,119,84,
    84,84,1399
84  1720 DATA 65,84,65,84,84,84,
    84,65,119,65,84,65,84,65,66,
    65,1228
8E  1730 DATA 84,84,84,65,119,65
    ,84,65,84,84,65,119,209,63,6
    4,66,1404
AE  1740 DATA 69,109,104,66,66,6
    6,128,65,66,65,66,66,65,66,1
    28,66,1261
FD  1750 DATA 66,65,65,66,65,66,
    65,66,128,65,65,66,66,65,66,
    66,1111
24  1760 DATA 128,66,66,66,209,6
    3,64,73,71,109,104,66,65,66,
    127,66,1409
83  1770 DATA 66,66,65,66,66,66,
    127,65,65,66,65,65,66,65,66,
    65,1110
C0  1780 DATA 127,65,66,65,66,66
    ,65,65,127,65,66,210,209,63,
    64,74,1463
04  1790 DATA 109,109,104,66,66,
    65,127,65,65,66,66,65,66,65,
    127,65,1296
32  1800 DATA 66,66,65,66,65,66,
    65,66,127,66,65,66,65,66,65,
    66,1111
5D  1810 DATA 127,66,65,66,209,6
    3,64,65,109,109,104,66,65,65
    ,66,65,1374
49  1820 DATA 66,66,66,65,65,66,
    65,66,65,65,66,65,66,65,66,6
    5,1048
2B  1830 DATA 66,65,66,65,66,65,
    66,65,66,65,66,65,209,63,66,
    66,1190
78  1840 DATA 109,109,104,66,66,
    66,66,171,187,170,65,66,65,6
    6,66,65,1507
B5  1850 DATA 66,66,65,66,65,65,
    66,66,66,65,66,66,65,66,171,
    218,1308
12  1860 DATA 170,65,65,66,209,6
    3,66,65,109,109,119,66,65,66
    ,65,187,1555
8D  1870 DATA 187,187,65,66,123,
    124,65,66,66,65,66,65,66,66,
    65,66,1408
11  1880 DATA 65,65,66,123,124,6
    5,211,218,211,66,66,65,209,6
    3,66,66,1749
FA  1890 DATA 109,109,120,66,66,
    65,66,187,187,187,65,65,125,
    126,65,65,1673
B9  1900 DATA 66,66,65,66,66,65,
    66,65,66,66,65,125,126,65,21
    1,218,1467
31  1910 DATA 211,66,65,66,209,6
    3,66,65,109,109,104,66,65,66
    ,65,187,1582
26  1920 DATA 187,187,65,66,66,6
```

```
     5,65,66,66,65,66,65,65,66,66
     ,66,1292
B3   1930 DATA 65,66,66,65,210,65
     ,211,218,211,66,66,65,209,63
     ,66,65,1777
68   1940 DATA 109,109,104,68,68,
     68,68,191,191,191,68,68,68,6
     8,68,68,1575
F3   1950 DATA 68,68,68,68,68,68,
     68,68,68,68,68,68,68,68,191,
     191,1334
46   1960 DATA 191,68,68,68,68,67
     ,66,66,0,255,0,255,0,255,27,
     16,1470
4D   1970 DATA 169,0,160,30,153,1
     95,2,136,16,250,160,14,153,6
     4,3,153,1658
05   1980 DATA 80,3,153,96,3,136,
     16,244,96,120,169,125,141,24
     ,3,169,1578
E9   1990 DATA 137,141,25,3,169,5
     4,133,1,88,169,8,32,210,255,
     169,147,1741
30   2000 DATA 32,210,255,32,193,
     143,96,169,231,141,148,65,14
     1,116,82,32,2086
91   2010 DATA 144,103,32,132,134
     ,32,0,134,32,145,135,32,165,
     135,32,176,1563
DF   2020 DATA 135,32,252,135,32,
     12,135,32,35,135,32,151,147,
     32,213,134,1644
CF   2030 DATA 32,51,135,96,120,1
     69,139,141,20,3,169,106,141,
     21,3,169,1515
68   2040 DATA 105,141,18,208,32,
     170,147,169,0,141,188,2,141,
     19,3,141,1625
FC   2050 DATA 112,3,88,96,169,24
     ,141,124,3,169,75,141,125,3,
     169,94,1536
A9   2060 DATA 141,122,3,169,72,1
     41,123,3,169,208,141,181,2,1
     69,218,141,2003
C1   2070 DATA 182,2,169,120,141,
     167,2,169,18,141,172,2,169,1
     ,141,21,1617
D6   2080 DATA 208,169,198,141,0,
     208,169,193,141,1,208,169,32
     ,141,170,2,2150
1E   2090 DATA 169,92,141,171,2,1
     69,45,141,185,2,169,75,141,1
     86,2,169,1859
36   2100 DATA 28,141,194,2,96,16
     9,16,141,187,2,141,189,2,10,
     141,137,1596
2C   2110 DATA 3,173,30,208,162,4
     ,142,246,2,142,232,2,202,142
     ,230,2,1922
83   2120 DATA 169,7,141,39,208,1
     41,40,208,169,0,141,27,208,1
     41,247,2,1888
1F   2130 DATA 141,32,208,141,37,
     208,169,10,141,38,208,96,169
     ,3,170,157,1928
E5   2140 DATA 54,3,157,58,3,202,
     16,247,169,0,162,5,157,41,20
     8,202,1684
C2   2150 DATA 16,250,96,169,0,16
     2,23,157,0,212,202,16,250,16
     9,15,141,1878
97   2160 DATA 24,212,96,169,206,
     141,184,2,169,10,141,183,2,1
     41,236,2,1918
FE   2170 DATA 169,198,141,235,2,
     169,220,141,133,3,169,218,14
     1,134,3,169,2245
10   2180 DATA 193,141,52,3,169,1
     77,141,53,3,162,3,142,63,3,1
     42,132,1579
FD   2190 DATA 3,142,113,3,142,19
     3,2,142,190,2,142,237,2,142,
     28,208,1691
B7   2200 DATA 160,209,152,157,24
     2,2,200,202,16,248,162,3,160
     ,213,152,157,2435
67   2210 DATA 238,2,200,202,16,2
     48,169,197,141,135,3,169,200
     ,141,136,3,2200
70   2220 DATA 96,169,8,162,0,157
     ,0,216,157,0,217,157,0,218,1
     57,232,1946
70   2230 DATA 218,232,208,241,96
     ,169,10,162,79,157,248,218,2
     02,16,250,96,2602
91   2240 DATA 169,24,141,191,2,1
     33,253,169,5,141,192,2,133,2
     54,169,112,2090
75   2250 DATA 133,251,169,72,133
     ,252,160,0,132,255,24,162,17
     ,177,251,145,2333
AE   2260 DATA 253,165,253,105,40
     ,133,253,144,3,230,254,24,23
     0,251,208,2,2548
35   2270 DATA 230,252,202,16,232
     ,230,255,165,255,201,40,176,
     14,109,191,2,2570
B0   2280 DATA 133,253,173,192,2,
     105,0,133,254,144,208,96,160
     ,0,162,39,2054
CE   2290 DATA 169,167,157,0,4,15
     7,80,4,157,160,4,157,240,4,1
     52,157,1769
0E   2300 DATA 0,216,157,80,216,1
     57,160,216,157,240,216,202,1
     6,226,162,5,2426
6A   2310 DATA 160,15,24,32,240,2
     55,162,0,189,103,137,240,6,3
     2,210,255,2060
02   2320 DATA 232,208,245,162,4,
     169,162,157,53,4,157,93,4,15
     7,133,4,1944
46   2330 DATA 157,64,4,157,104,4
     ,157,144,4,202,16,235,169,10
     0,162,5,1684
16   2340 DATA 157,58,4,157,98,4,
     157,138,4,202,16,244,141,92,
     4,141,1617
1A   2350 DATA 40,4,141,120,4,141
     ,200,4,141,109,4,162,169,142
     ,100,4,1485
AE   2360 DATA 202,142,101,4,169,
     144,32,210,255,160,1,162,1,2
     4,32,240,1879
7B   2370 DATA 255,202,189,14,137
     ,240,6,32,210,255,232,208,24
     5,160,29,162,2576
6C   2380 DATA 1,24,32,240,255,20
     2,189,29,137,240,6,32,210,25
     5,232,208,2292
A0   2390 DATA 245,160,1,162,3,24
     ,32,240,255,162,0,189,43,137
     ,240,6,1899
23   2400 DATA 32,210,255,232,208
     ,245,160,29,162,3,24,32,240,
     255,162,0,2249
13   2410 DATA 189,57,137,240,6,3
     2,210,255,232,208,245,160,1,
     162,5,24,2163
1B   2420 DATA 32,240,255,162,0,1
     89,71,137,240,6,32,210,255,2
     32,208,245,2514
D4   2430 DATA 160,27,162,5,24,32
     ,240,255,162,0,189,87,137,32
     ,210,255,1977
3C   2440 DATA 240,3,232,208,245,
     162,17,169,8,157,92,216,157,
     172,216,202,2496
17   2450 DATA 16,247,162,4,189,1
     35,206,157,127,4,202,16,247,
     96,18,83,1909
67   2460 DATA 67,79,82,69,58,146
     ,48,48,48,48,48,164,0,164,18
     ,75,1162
24   2470 DATA 69,89,83,32,58,146
     ,164,164,18,58,0,66,69,83,84
     ,32,1215
76   2480 DATA 58,146,32,32,32,32
     ,32,164,0,164,18,76,73,86,69
     ,83,1097
BE   2490 DATA 58,146,48,55,18,58
     ,0,84,73,77,69,32,58,146,49,
     48,1019
16   2500 DATA 49,48,48,164,164,1
     64,0,164,164,164,18,66,79,76
     ,84,83,1535
FE   2510 DATA 58,146,49,50,48,32
     ,0,18,28,87,73,76,76,73,65,7
     7,956
5C   2520 DATA 32,84,69,76,76,146
     ,0,32,195,147,32,25,134,32,1
     62,144,1386
17   2530 DATA 32,168,229,32,151,
     147,32,0,122,169,0,141,62,3,
     240,115,1643
15   2540 DATA 32,168,229,32,195,
     147,32,55,134,32,204,147,32,
     100,134,173,1846
E5   2550 DATA 96,3,208,6,32,67,1
     39,32,164,146,173,209,2,240,
     3,32,1552
DF   2560 DATA 30,140,173,206,2,2
     40,3,32,114,138,173,19,3,208
     ,56,173,1710
1D   2570 DATA 221,2,208,246,32,1
     91,138,173,112,3,240,211,173
     ,172,2,208,2332
DF   2580 DATA 206,173,167,2,240,
     4,201,240,208,197,32,195,147
     ,32,56,138,2238
03   2590 DATA 32,162,144,32,0,16
     0,169,147,32,210,255,32,151,
     147,32,145,1850
80   2600 DATA 135,32,103,142,76,
     3,138,32,102,138,32,195,147,
     32,162,144,1613
6F   2610 DATA 32,100,142,173,0,2
     20,41,16,240,37,173,229,2,20
     8,244,173,2030
47   2620 DATA 62,3,73,1,141,62,3
     ,208,3,76,125,137,169,48,160
     ,4,1275
58   2630 DATA 153,159,206,136,16
     ,250,32,162,144,32,41,143,76
     ,3,138,32,1723
89   2640 DATA 162,144,32,151,147
     ,76,144,137,162,4,189,47,4,1
     57,0,205,1761
C7   2650 DATA 189,127,4,157,6,20
     5,189,207,4,157,12,205,202,1
     6,235,162,2077
BB   2660 DATA 1,189,156,4,157,16
     ,205,202,16,247,162,2,189,23
     6,4,157,1943
6E   2670 DATA 19,205,202,16,247,
     96,162,4,189,127,4,157,115,3
     ,202,16,1764
D9   2680 DATA 247,96,162,0,142,2
     06,2,142,141,3,142,142,3,232
     ,142,143,1945
99   2690 DATA 3,232,56,189,236,4
     ,253,141,3,201,48,176,2,105,
     10,157,1816
7D   2700 DATA 236,4,202,16,238,2
     32,142,11,212,142,13,212,169
     ,9,141,12,1991
F3   2710 DATA 212,169,3,141,8,21
     2,169,65,141,11,212,162,1,14
     2,10,212,1870
3D   2720 DATA 232,189,236,4,201,
     48,208,6,202,16,246,141,247,
     2,96,173,2247
DB   2730 DATA 198,2,208,58,174,1
     67,2,224,120,208,52,172,172,
     2,240,18,2017
02   2740 DATA 192,18,240,28,169,
     100,141,102,4,141,103,4,141,
     98,4,141,1626
```

```
DE   2750 DATA 99,4,169,172,141,6
     0,4,169,173,141,61,4,192,0,2
     40,15,1644
93   2760 DATA 169,174,141,140,4,
     169,175,141,141,4,192,18,240
     ,13,96,169,1986
D1   2770 DATA 100,141,140,4,141,
     141,4,224,120,240,22,169,100
     ,141,60,4,1751
D2   2780 DATA 141,61,4,224,0,208
     ,10,169,100,141,102,4,141,10
     3,4,208,1620
F1   2790 DATA 23,169,167,141,102
     ,4,169,164,141,103,4,224,240
     ,208,9,169,2037
A9   2800 DATA 100,141,98,4,141,9
     9,4,96,169,163,141,98,4,169,
     167,141,1735
CF   2810 DATA 99,4,96,173,30,208
     ,141,130,3,41,1,208,9,169,2,
     45,1359
D4   2820 DATA 130,3,240,21,208,6
     0,169,198,141,248,7,141,96,3
     ,141,234,2040
2B   2830 DATA 2,169,9,141,5,212,
     141,6,212,96,169,253,45,21,2
     08,141,1830
F9   2840 DATA 21,208,169,0,141,6
     6,3,141,204,2,141,205,2,141,
     82,3,1529
EF   2850 DATA 138,74,168,169,219
     ,153,248,7,157,96,3,141,209,
     2,173,30,1987
3E   2860 DATA 208,96,162,6,173,2
     15,2,240,2,202,202,189,56,11
     2,45,21,1931
38   2870 DATA 208,240,13,189,96,
     3,208,8,189,56,112,45,130,3,
     208,7,1715
24   2880 DATA 232,232,224,14,208
     ,229,96,173,249,7,201,199,24
     0,28,173,16,2521
85   2890 DATA 208,61,56,112,208,
     234,56,173,2,208,233,8,221,0
     ,208,176,2164
AC   2900 DATA 223,105,16,221,0,2
     08,176,146,144,214,173,16,20
     8,61,56,112,2079
C7   2910 DATA 208,31,169,2,45,16
     ,208,208,199,56,173,2,208,23
     3,8,221,1987
D6   2920 DATA 0,208,176,188,105,
     16,144,2,176,220,221,0,208,1
     44,177,176,2161
5E   2930 DATA 213,24,169,2,45,16
     ,208,240,12,173,2,208,105,8,
     221,0,1646
4E   2940 DATA 208,144,157,176,19
     3,173,2,208,105,8,144,148,17
     6,240,169,0,2251
A3   2950 DATA 141,209,2,141,141,
     3,141,142,3,141,143,3,169,2,
     141,144,1666
C5   2960 DATA 3,162,5,142,145,3,
     202,24,189,47,4,125,141,3,20
     1,58,1454
0E   2970 DATA 144,2,233,10,157,4
     7,4,202,16,238,232,189,127,4
     ,221,47,1873
2E   2980 DATA 4,240,4,176,7,144,
     6,232,224,4,208,239,96,173,1
     31,4,1892
3C   2990 DATA 105,5,201,58,144,2
     ,233,10,141,131,4,173,130,4,
     105,2,1448
6C   3000 DATA 201,58,144,2,233,1
     0,141,130,4,162,2,189,127,4,
     105,0,1512
62   3010 DATA 201,58,144,2,233,1
     0,157,127,4,202,16,239,96,32
     ,151,147,1819
52   3020 DATA 169,147,32,210,255
     ,32,144,103,32,35,135,32,213
     ,144,32,147,1862
0F   3030 DATA 146,162,13,142,248
     ,7,232,142,249,7,232,142,250
     ,7,169,192,2340
21   3040 DATA 141,4,208,169,184,
     141,2,208,169,117,141,3,208,
     141,5,208,2049
62   3050 DATA 169,7,141,21,208,1
     62,0,142,63,3,142,16,208,142
     ,39,208,1671
12   3060 DATA 142,40,208,142,41,
     208,142,28,208,142,32,208,16
     9,8,141,33,1892
78   3070 DATA 208,32,147,135,162
     ,38,169,14,157,144,217,202,1
     6,250,162,120,2173
CD   3080 DATA 169,13,157,223,217
     ,202,208,250,162,25,169,1,15
     7,232,217,202,2604
61   3090 DATA 16,250,169,15,141,
     85,219,141,124,219,169,12,14
     1,100,219,141,2161
84   3100 DATA 139,219,162,0,189,
     16,130,157,0,4,189,16,131,15
     7,0,5,1514
9C   3110 DATA 189,16,132,157,0,6
     ,189,248,132,157,232,6,232,2
     08,229,169,2302
11   3120 DATA 75,141,123,5,141,1
     27,5,169,45,141,124,5,141,12
     5,5,141,1513
AE   3130 DATA 126,5,169,0,133,2,
     141,253,207,32,204,147,32,18
     8,144,169,1952
75   3140 DATA 0,141,245,217,173,
     0,220,41,16,208,249,165,3,20
     8,245,32,2163
65   3150 DATA 162,144,169,208,13
     3,253,169,207,133,254,162,25
     ,160,8,136,48,2371
9A   3160 DATA 25,185,112,56,209,
     253,240,246,202,48,15,56,165
     ,253,233,8,2306
63   3170 DATA 133,253,176,232,19
     8,254,160,7,208,231,138,189,
     136,144,164,2,2625
72   3180 DATA 153,150,206,32,15,
     142,169,15,141,24,212,169,10
     0,141,254,207,2130
AA   3190 DATA 24,173,245,217,72,
     105,1,41,7,141,245,217,104,7
     3,32,141,1838
D1   3200 DATA 1,212,206,253,207,
     208,233,206,254,207,208,228,
     160,7,162,0,2752
89   3210 DATA 189,112,56,72,189,
     113,56,157,112,56,232,224,7,
     208,245,104,2132
55   3220 DATA 157,112,56,32,34,1
     42,169,16,141,254,207,206,25
     3,207,208,251,2445
05   3230 DATA 206,254,207,208,24
     6,152,42,42,42,42,141,1,212,
     136,16,206,2153
A1   3240 DATA 169,192,141,0,208,
     169,137,141,1,208,32,194,145
     ,230,2,165,2134
EA   3250 DATA 2,201,3,240,6,32,1
     88,144,76,79,141,32,235,144,
     96,169,1788
D6   3260 DATA 0,141,4,212,169,10
     ,141,5,212,141,6,212,169,17,
     141,4,1584
4E   3270 DATA 212,96,169,16,141,
     4,212,169,15,141,5,212,169,1
     3,141,6,1721
3E   3280 DATA 212,169,21,141,1,2
     12,169,17,141,4,212,96,169,0
     ,141,4,1709
F6   3290 DATA 212,169,6,141,5,21
     2,141,6,212,169,17,141,4,212
     ,96,169,1912
FD   3300 DATA 32,141,11,212,169,
     15,141,12,212,169,13,141,13,
     212,169,33,1695
4E   3310 DATA 141,11,212,96,32,1
     23,142,32,135,142,176,11,32,
     141,140,32,1598
93   3320 DATA 155,142,169,0,141,
     21,208,32,41,143,96,160,4,18
     5,47,4,1548
BB   3330 DATA 153,159,206,136,16
     ,247,96,160,0,185,9,206,217,
     159,206,240,2395
4E   3340 DATA 3,176,7,96,200,192
     ,5,208,240,56,96,169,135,133
     ,253,169,2138
EA   3350 DATA 206,133,254,160,0,
     177,253,217,159,206,240,4,17
     6,8,144,19,2356
D1   3360 DATA 200,192,5,208,240,
     56,165,253,233,14,133,253,17
     6,229,198,254,2809
17   3370 DATA 76,163,142,56,165,
     253,233,9,141,141,3,165,254,
     233,0,141,2175
AD   3380 DATA 142,3,169,14,133,2
     51,169,206,133,252,169,0,133
     ,253,169,206,2402
AA   3390 DATA 133,254,160,13,177
     ,251,145,253,136,16,249,165,
     253,205,141,3,2554
13   3400 DATA 144,7,165,254,205,
     142,3,240,27,24,165,251,105,
     14,133,251,2130
94   3410 DATA 165,252,105,0,133,
     252,165,253,105,14,133,253,1
     65,254,105,0,2354
8E   3420 DATA 133,254,144,206,17
     3,141,3,133,253,173,142,3,13
     3,254,160,13,2318
1A   3430 DATA 185,150,206,145,25
     3,136,16,248,96,32,195,147,1
     69,147,32,210,2367
C1   3440 DATA 255,32,213,147,32,
     151,147,32,145,135,162,1,142
     ,62,3,202,1861
9F   3450 DATA 189,128,126,157,0,
     4,189,128,127,157,0,5,189,12
     8,128,157,1812
91   3460 DATA 0,6,189,200,128,15
     7,72,6,232,208,229,169,6,141
     ,35,208,1986
C9   3470 DATA 169,9,141,33,208,2
     32,142,34,208,142,134,2,232,
     160,4,24,1874
31   3480 DATA 32,240,255,160,0,1
     85,40,144,240,6,32,210,255,2
     00,208,245,2452
1F   3490 DATA 169,126,133,253,16
     9,206,133,254,169,9,133,255,
     162,4,160,4,2339
29   3500 DATA 24,32,240,255,160,
     0,177,253,32,210,255,200,192
     ,14,208,246,2498
D9   3510 DATA 198,255,16,15,169,
     2,141,229,2,32,230,146,32,20
     4,147,32,1850
57   3520 DATA 99,147,96,56,165,2
     53,233,14,133,253,176,2,198,
     254,232,208,2519
49   3530 DATA 205,169,45,160,140
     ,153,255,205,136,208,250,169
     ,0,133,253,169,2650
A4   3540 DATA 206,133,254,162,0,
     160,0,189,56,144,145,253,232
     ,200,192,3,2329
FB   3550 DATA 208,245,224,30,176
     ,14,165,253,105,14,133,253,1
     65,254,105,0,2344
E2   3560 DATA 133,254,144,225,16
     9,9,133,253,169,206,133,254,
     162,0,160,0,2404
15   3570 DATA 189,86,144,145,253
     ,232,200,192,5,208,245,224,5
     0,176,14,165,2528
```

```
86  3580 DATA 253,105,14,133,253
    ,165,254,105,0,133,254,144,2
    25,169,45,160,2412
3D  3590 DATA 5,153,153,206,136,
    16,250,96,5,84,79,80,32,84,6
    9,78,1526
A2  3600 DATA 32,83,67,79,82,69,
    83,0,68,87,76,68,87,76,68,87
    ,1112
55  3610 DATA 76,68,87,76,68,87,
    76,68,87,76,68,87,76,68,87,7
    6,1231
A5  3620 DATA 68,87,76,68,87,76,
    48,49,48,53,48,48,50,50,48,5
    3,957
A6  3630 DATA 48,50,52,53,48,48,
    51,54,48,48,48,51,56,50,53,4
    8,806
5E  3640 DATA 52,49,55,48,48,52,
    51,48,48,48,53,53,48,48,48,5
    3,802
35  3650 DATA 55,55,53,48,54,50,
    48,48,65,66,67,68,69,70,71,7
    2,959
8F  3660 DATA 73,74,75,76,77,78,
    79,80,81,82,83,84,85,86,87,8
    8,1288
1F  3670 DATA 89,90,120,169,49,1
    41,20,3,169,234,141,21,3,173
    ,14,220,1656
1C  3680 DATA 9,1,141,14,220,169
    ,240,141,26,208,88,96,120,16
    9,247,141,2030
48  3690 DATA 20,3,169,144,141,2
    1,3,169,152,141,18,208,32,17
    0,147,169,1707
FC  3700 DATA 0,133,3,88,96,160,
    216,185,255,55,153,255,206,1
    36,208,247,2396
5F  3710 DATA 160,7,169,0,153,0,
    56,136,16,250,96,160,216,185
    ,255,206,2065
B0  3720 DATA 153,255,55,136,208
    ,247,96,173,25,208,141,25,20
    8,41,1,208,2180
CF  3730 DATA 3,76,49,234,169,15
    2,141,18,208,173,17,208,41,1
    27,141,17,1774
0C  3740 DATA 208,165,3,240,7,17
    3,255,207,240,17,208,101,173
    ,0,220,41,2258
7A  3750 DATA 15,73,15,201,4,240
    ,90,201,8,208,83,162,7,189,8
    ,56,1560
FD  3760 DATA 41,128,240,2,169,1
    ,157,247,207,202,16,241,160,
    0,140,255,2206
D6  3770 DATA 207,185,8,56,10,13
    3,255,185,16,56,16,2,230,255
    ,165,255,2034
43  3780 DATA 153,8,56,200,192,2
    08,208,233,162,7,189,247,207
    ,29,208,56,2363
6B  3790 DATA 157,208,56,202,16,
    244,24,165,3,105,1,41,7,72,1
    09,1,1411
A8  3800 DATA 212,141,1,212,104,
    133,3,201,7,208,3,32,34,142,
    76,49,1558
7D  3810 DATA 234,162,7,142,255,
    207,189,208,56,41,1,240,2,16
    9,128,157,2198
EE  3820 DATA 247,207,202,16,241
    ,160,208,185,7,56,74,133,255
    ,185,255,55,2486
60  3830 DATA 41,1,240,6,165,255
    ,9,128,208,2,165,255,153,7,5
    6,136,1827
99  3840 DATA 208,229,162,7,189,
    247,207,29,8,56,157,8,56,202
    ,16,244,2025
F4  3850 DATA 48,164,160,0,162,0
    ,189,112,56,153,64,3,200,200
    ,200,232,1943
19  3860 DATA 224,8,144,242,169,
    5,141,253,207,173,18,208,208
    ,251,206,253,2710
19  3870 DATA 207,208,246,173,1,
    208,72,141,1,212,32,60,142,1
    04,56,233,2096
8E  3880 DATA 1,141,1,208,201,11
    7,208,220,173,63,3,208,30,16
    2,62,189,1987
E0  3890 DATA 64,3,157,128,3,202
    ,16,247,162,62,169,0,157,64,
    3,202,1639
97  3900 DATA 16,250,169,0,141,4
    ,212,238,63,3,96,201,2,176,1
    3,162,1746
C3  3910 DATA 62,189,64,3,157,19
    2,3,202,16,247,48,220,24,173
    ,0,208,1808
14  3920 DATA 105,8,141,0,208,16
    9,5,141,253,207,173,18,208,2
    08,251,206,2301
E2  3930 DATA 253,207,208,246,23
    8,39,208,238,40,208,238,41,2
    08,206,2,208,2788
2D  3940 DATA 206,3,208,24,173,0
    ,208,105,1,144,10,72,173,16,
    208,9,1560
F8  3950 DATA 1,141,16,208,104,1
    41,0,208,206,1,208,56,173,5,
    208,233,1909
6F  3960 DATA 1,201,40,144,157,1
    41,5,208,233,40,141,1,212,16
    9,0,141,1834
F2  3970 DATA 4,212,169,6,141,5,
    212,169,10,141,6,212,169,129
    ,141,4,1730
BD  3980 DATA 212,208,162,162,62
    ,169,0,157,64,3,157,128,3,15
    7,192,3,1839
A5  3990 DATA 202,16,244,96,173,
    172,2,240,60,169,32,205,110,
    6,240,53,2020
93  4000 DATA 172,1,208,192,173,
    176,46,174,167,2,224,3,176,5
    ,141,116,1976
0E  4010 DATA 82,208,7,224,237,1
    44,30,141,148,65,141,110,6,3
    2,79,142,1796
F8  4020 DATA 162,0,169,236,205,
    76,4,208,4,232,142,112,3,157
    ,76,4,1790
7E  4030 DATA 169,180,141,224,2,
    96,160,7,140,134,2,162,39,16
    9,32,157,1814
A3  4040 DATA 72,7,152,157,72,21
    9,202,16,244,232,142,134,2,1
    62,119,169,2101
22  4050 DATA 100,157,112,7,202,
    16,250,162,21,160,10,24,32,2
    40,255,169,1917
9D  4060 DATA 7,141,134,2,162,0,
    189,131,147,240,6,32,210,255
    ,232,208,2096
B6  4070 DATA 245,96,173,25,208,
    141,25,208,41,1,240,52,169,2
    24.141.18.2007
44  4080 DATA 208,173,17,208,41,
    127,141,17,208,162,0,173,72,
    7,72,189,1815
8A  4090 DATA 73,7,157,72,7,232,
    224,40,208,245,104,141,111,7
    ,206,253,2087
0F  4100 DATA 207,208,13,206,254
    ,207,208,8,169,3,141,254,207
    ,206,229,2,2522
3F  4110 DATA 76,49,234,120,169,
    34,141,20,3,169,147,141,21,3
    ,169,224,1720
72  4120 DATA 141,18,208,141,253
    ,207,169,3,141,254,207,32,21
    3,147,32,170,2336
EB  4130 DATA 147,88,96,80,82,69
    ,83,83,32,70,73,82,69,32,84,
    79,1249
3A  4140 DATA 32,83,84,65,82,84,
    0,169,31,141,24,208,173,17,2
    08,9,1410
7F  4150 DATA 27,141,17,208,169,
    216,141,22,208,96,173,17,208
    ,41,127,141,1952
2B  4160 DATA 17,208,173,26,208,
    9,1,141,26,208,173,14,220,41
    ,254,141,1860
7D  4170 DATA 14,220,96,173,17,2
    08,41,239,141,17,208,96,173,
    17,208,9,1877
F7  4180 DATA 16,141,17,208,96,1
    69,0,141,21,208,141,23,208,1
    41,27,208,1765
76  4190 DATA 141,29,208,162,16,
    157,0,208,202,16,250,96,0,0,
    0,0,1485
E2  4200 DATA 169,0,133,250,169,
    122,133,251,169,0,133,174,13
    3,193,169,122,2320
12  4210 DATA 133,175,133,194,16
    9,240,133,252,169,147,133,25
    3,160,0,177,250,2718
F7  4220 DATA 145,174,230,250,20
    8,2,230,251,230,174,208,2,23
    0,175,165,250,2924
73  4230 DATA 197,252,208,234,16
    5,251,197,253,208,228,169,13
    9,133,187,169,148,3138
47  4240 DATA 133,188,169,3,133,
    183,169,0,133,185,160,0,185,
    91,148,240,2120
F0  4250 DATA 6,32,210,255,200,2
    08,245,32,207,255,240,251,20
    1,49,240,4,2635
ED  4260 DATA 201,56,48,230,41,1
    5,133,186,76,234,245,147,17,
    17,73,78,1797
F0  4270 DATA 80,85,84,32,68,69,
    86,73,67,69,32,78,85,77,66,6
    9,1120
F8  4280 DATA 82,13,17,67,65,83,
    61,49,32,47,32,68,73,83,75,6
    1,908
C7  4290 DATA 32,56,32,79,82,32,
    57,58,45,32,0,87,84,53,0,0,7
    29
0C  4300 DATA 0,0,0,0,0,0,0,0,0,
    0,0,255,255,255,255,0,1020
```

PROGRAM:WILLIAM TELL - LISTING 6

```
D4  10 BL=372   :LN=50    :SA=3404
    8
F9  20 FOR L=0 TO BL:CX=0:FOR D=
    0 TO 15
8B  21 READ A:POKE53280,A
9E  22 CX=CX+A:POKE SA+L*16+D,A:
    NEXT
A2  30 READA:IF A=CX THEN40
12  31 PRINT"ERROR IN LINE";LN+(
    L*10):STOP
06  40 NEXT L:SYS39824
9E  50 DATA 32,195,147,32,35,161
    ,32,35,135,32,216,162,32,184
    ,160,169,1759
D1  60 DATA 90,32,207,160,32,204
    ,147,32,158,160,173,137,3,24
    0,251,32,2058
03  70 DATA 89,173,173,206,2,240
    ,3,32,235,168,173,207,2,240,
    6,32,1981
0A  80 DATA 140,167,32,67,169,17
    3,209,2,240,9,32,116,167,32,
    108,169,1832
59  90 DATA 32,164,169,173,216,2
    ,240,3,32,164,169,173,221,2,
    208,26,1994
```

```
BE  100 DATA 173,202,2,240,15,32
    ,208,179,32,162,144,32,140,1
    60,169,1,1891
EB  110 DATA 141,221,2,96,32,8,1
    74,76,26,160,32,140,160,32,1
    62,144,1606
32  120 DATA 169,1,141,21,208,17
    3,18,208,208,251,238,33,208,
    238,39,208,2362
A1  130 DATA 32,248,108,238,1,20
    8,208,237,32,0,36,96,162,4,1
    89,8,1807
F5  140 DATA 4,157,159,206,189,4
    8,4,157,115,3,202,16,241,96,
    120,169,1886
3D  150 DATA 48,141,18,208,169,5
    4,141,20,3,169,163,141,21,3,
    169,0,1468
E7  160 DATA 141,188,2,32,170,14
    7,88,96,162,63,169,0,157,216
    ,34,202,1867
46  170 DATA 16,250,96,162,63,18
    9,24,35,157,216,34,202,16,24
    7,96,141,1944
39  180 DATA 154,30,141,174,30,1
    41,194,30,141,214,30,141,234
    ,30,141,254,2079
CD  190 DATA 30,141,18,31,141,38
    ,31,141,58,31,141,78,31,141,
    98,31,1180
9D  200 DATA 141,118,31,141,138,
    31,141,158,31,141,178,31,141
    ,198,31,141,1791
2B  210 DATA 218,31,141,238,31,9
    6,162,39,169,90,221,24,5,208
    ,5,169,1847
E5  220 DATA 75,157,24,5,202,16,
    241,96,160,13,162,9,140,35,2
    08,142,1685
F8  230 DATA 34,208,96,169,147,3
    2,210,255,32,105,172,32,24,1
    61,169,13,1859
95  240 DATA 32,147,135,169,24,1
    41,24,208,169,200,141,191,2,
    133,253,169,2138
E8  250 DATA 4,141,192,2,133,254
    ,173,167,2,240,48,162,0,142,
    167,2,1829
92  260 DATA 169,244,141,122,3,1
    69,28,141,123,3,169,0,141,12
    4,3,169,1749
56  270 DATA 32,141,125,3,169,8,
    133,251,169,29,133,252,169,2
    08,141,189,2152
D3  280 DATA 2,9,7,141,187,2,141
    ,22,208,208,44,162,202,142,1
    67,2,1646
DD  290 DATA 169,44,141,122,3,16
    9,13,141,123,3,169,56,141,12
    4,3,169,1590
57  300 DATA 16,141,125,3,169,64
    ,133,251,169,13,133,252,169,
    208,141,189,2176
6A  310 DATA 2,141,187,2,141,22,
    208,169,0,133,255,168,24,162
    ,19,177,1810
2C  320 DATA 251,145,253,165,253
    ,105,40,133,253,144,3,230,25
    4,24,230,251,2734
D3  330 DATA 208,2,230,252,202,1
    6,232,230,255,165,255,201,40
    ,176,14,109,2587
A9  340 DATA 191,2,133,253,173,1
    92,2,105,0,133,254,144,208,1
    69,2,141,2102
19  350 DATA 190,2,162,0,189,63,
    172,157,1,4,189,70,172,157,4
    1,4,1573
E3  360 DATA 189,77,172,157,81,4
    ,189,84,172,157,27,4,189,91,
    172,157,1922
FD  370 DATA 67,4,189,98,172,157
    ,107,4,232,224,7,208,215,169
    ,48,141,2042
63  380 DATA 35,4,141,36,4,162,1
    1,169,1,157,2,216,157,42,216
    ,157,1510
30  390 DATA 82,216,202,16,244,1
    62,9,157,28,216,157,68,216,1
    57,108,216,2254
E2  400 DATA 202,16,244,169,65,1
    41,14,4,141,54,4,141,94,4,14
    1,38,1472
D2  410 DATA 4,141,78,4,141,118,
    4,169,108,162,14,157,120,4,2
    02,16,1442
0C  420 DATA 250,162,11,157,147,
    4,202,16,250,169,109,141,146
    ,4,162,4,1934
15  430 DATA 189,0,205,157,8,4,1
    89,6,205,157,48,4,189,12,205
    ,157,1735
A9  440 DATA 88,4,202,16,235,162
    ,2,189,19,205,157,114,4,202,
    16,247,1862
D4  450 DATA 162,1,189,16,205,15
    7,75,4,202,16,247,162,4,142,
    183,2,1767
A6  460 DATA 142,52,3,202,142,23
    7,2,142,236,2,142,28,208,162
    ,11,142,1853
55  470 DATA 37,208,202,142,38,2
    08,169,7,141,39,208,169,1,14
    1,40,208,1958
AE  480 DATA 169,16,141,193,2,16
    9,198,141,235,2,169,206,141,
    184,2,162,2130
7D  490 DATA 3,160,193,152,157,1
    73,2,200,202,16,248,162,3,16
    0,203,152,2186
F8  500 DATA 157,177,2,136,202,1
    6,248,96,162,1,142,225,2,142
    ,226,2,1936
E7  510 DATA 202,142,32,208,142,
    33,208,142,195,2,142,194,2,1
    42,197,2,1985
7C  520 DATA 142,196,2,142,213,2
    ,142,205,2,142,204,2,142,135
    ,3,142,1816
37  530 DATA 136,3,142,66,3,142,
    82,3,142,137,3,142,138,3,142
    ,139,1423
F2  540 DATA 3,142,207,2,142,206
    ,2,142,216,2,142,203,2,142,1
    99,2,1754
DC  550 DATA 142,220,2,142,221,2
    ,142,218,2,142,200,2,142,222
    ,2,142,1943
BD  560 DATA 202,2,142,208,2,96,
    173,25,208,141,25,208,41,1,2
    40,35,1749
17  570 DATA 173,17,208,41,127,1
    41,17,208,173,188,2,240,3,76
    ,87,164,1865
84  580 DATA 174,187,2,142,22,20
    8,169,250,141,18,208,206,188
    ,2,173,137,2227
12  590 DATA 3,208,3,76,49,234,1
    73,205,2,240,6,32,195,167,76
    ,121,1790
48  600 DATA 163,173,204,2,240,3
    ,32,41,168,173,203,2,208,229
    ,32,177,2050
B9  610 DATA 168,174,167,2,224,2
    02,240,51,173,225,2,208,46.2
    38,187,2,2309
D4  620 DATA 173,187,2,41,8,240,
    36,162,19,169,200,133,253,16
    9,4,133,1929
4F  630 DATA 254,169,201,133,251
    ,169,4,133,252,169,40,133,20
    ,169,20,133,2250
78  640 DATA 21,32,188,103,169,1
    6,141,187,2,208,63,174,167,2
    ,208,11,1692
44  650 DATA 173,222,2,240,53,32
    ,253,163,76,49,234,173,226,2
    ,208,42,2148
32  660 DATA 173,187,2,41,7,208,
    32,162,19,169,200,133,253,16
    9,4,133,1892
39  670 DATA 254,169,199,133,251
    ,169,4,133,252,169,20,133,20
    ,32,42,104,2084
5D  680 DATA 169,23,141,187,2,20
    8,3,206,187,2,76,49,234,173,
    208,2,1870
B3  690 DATA 208,11,173,1,208,20
    1,100,176,3,141,208,2,96,24,
    173,0,1725
87  700 DATA 208,105,2,144,10,72
    ,173,16,208,9,1,141,16,208,1
    04,141,1558
0A  710 DATA 0,208,206,237,2,208
    ,229,238,136,3,169,2,141,237
    ,2,169,2187
75  720 DATA 3,45,136,3,141,136,
    3,170,189,177,2,141,248,7,22
    4,0,1625
D9  730 DATA 208,202,32,248,108,
    173,16,208,41,1,240,192,173,
    0,208,201,2251
BB  740 DATA 100,144,185,141,202
    ,2,96,174,189,2,142,22,208,2
    38,188,2,2035
53  750 DATA 169,88,141,18,208,3
    2,40,170,173,208,2,208,30,17
    3,203,2,1865
A1  760 DATA 240,28,173,218,2,20
    8,14,32,35,135,32,17,173,173
    ,221,2,1703
57  770 DATA 240,3,76,129,234,32
    ,198,117,32,119,170,76,129,2
    34,173,137,2099
7E  780 DATA 3,208,6,32,25,171,7
    6,129,234,32,129,174,162,0,3
    2,203,1616
5F  790 DATA 166,160,41,169,62,2
    09,251,208,47,169,32,141,209
    ,2,145,251,2262
7F  800 DATA 169,144,133,2,169,1
    ,133,3,24,165,2,109,122,3,13
    3,2,1314
95  810 DATA 165,3,109,123,3,133
    ,3,160,20,169,62,209,2,240,3
    .136.1540
E2  820 DATA 16,249,169,32,145,2
    ,208,52,169,59,160,41,209,25
    1,208,16,1986
8D  830 DATA 169,32,145,251,141,
    203,2,141,99,3,32,116,165,76
    ,129,234,1938
3C  840 DATA 169,88,160,41,209,2
    51,240,5,200,209,251,208,15,
    160,41,169,2416
EE  850 DATA 32,145,251,200,145,
    251,200,145,251,141,207,2,17
    3,194,2,240,2579
94  860 DATA 19,32,51,167,173,19
    7,2,240,3,76,48,166,173,196,
    2,240,1785
B1  870 DATA 80,76,143,166,162,3
    ,160,80,189,93,167,209,251,2
    40,78,200,2297
D8  880 DATA 209,251,240,73,200,
    209,251,240,68,200,209,251,2
    40,63,202,16,2922
17  890 DATA 229,173,195,2,208,1
    0,160,81,32,99,167,144,48,32
    ,116,165,1861
B4  900 DATA 238,1,208,238,1,208
    ,173,195,2,240,10,206,195,2,
    240,5,2162
5A  910 DATA 206,195,2,208,12,17
    3,1,208,73,255,74,141,1,212,
    32,19,1812
B5  920 DATA 170,76,129,234,169,
    1,141,225,2,141,226,2,96,169
    ,0,141,1922
```

```
B5  930 DATA 195,2,173,0,220,41,
    15,73,15,133,255,208,43,32,1
    16,165,1686
7A  940 DATA 173,235,2,141,248,7
    ,206,236,2,208,214,173,213,2
    ,73,1,2134
61  950 DATA 141,213,2,208,3,32,
    248,108,169,10,141,236,2,173
    ,235,2,1923
F2  960 DATA 73,24,141,235,2,76,
    129,234,165,255,201,1,208,14
    ,32,97,1887
3A  970 DATA 167,176,202,32,116,
    165,32,155,166,76,129,234,20
    1,2,208,16,2077
EE  980 DATA 160,81,32,99,167,17
    6,222,32,116,165,32,181,166,
    76,129,234,2068
52  990 DATA 201,4,240,23,201,5,
    208,84,162,1,142,194,2,142,1
    97,2,1808
C2  1000 DATA 202,142,196,2,169,
    192,141,248,7,208,186,173,20
    4,2,208,15,2295
D4  1010 DATA 173,247,2,208,10,1
    73,0,220,41,16,208,3,32,195,
    167,206,1901
C4  1020 DATA 237,2,208,28,238,1
    35,3,169,3,141,237,2,45,135,
    3,141,1727
CB  1030 DATA 135,3,170,189,173,
    2,141,248,7,224,0,208,3,32,2
    48,108,1891
B6  1040 DATA 162,1,142,226,2,20
    2,142,225,2,76,129,234,201,8
    ,240,26,2018
C1  1050 DATA 201,9,240,3,76,141
    ,165,162,1,142,194,2,142,196
    ,2,202,1878
E1  1060 DATA 142,197,2,169,204,
    141,248,7,208,223,173,205,2,
    208,15,173,2317
10  1070 DATA 247,2,208,10,173,0
    ,220,41,16,208,3,32,41,168,2
    06,237,1812
47  1080 DATA 2,208,28,238,136,3
    ,169,3,141,237,2,45,136,3,14
    1,136,1628
D2  1090 DATA 3,170,189,177,2,14
    1,248,7,224,0,208,3,32,248,1
    08,162,1922
53  1100 DATA 1,142,225,2,202,14
    2,226,2,76,129,234,206,52,3,
    208,20,1870
86  1110 DATA 32,183,167,32,248,
    108,56,173,1,208,233,8,141,1
    ,208,169,1968
7F  1120 DATA 6,141,52,3,96,206,
    52,3,208,250,32,183,167,32,2
    48,108,1787
D6  1130 DATA 24,173,1,208,105,8
    ,141,1,208,208,228,169,0,133
    ,251,169,2027
32  1140 DATA 4,133,252,133,3,16
    9,0,133,254,173,16,208,61,56
    ,112,240,1947
57  1150 DATA 11,24,169,32,101,2
    51,133,251,144,2,230,252,189
    ,0,208,74,2071
2E  1160 DATA 74,74,56,233,3,24,
    101,251,133,251,24,189,1,208
    ,105,2,1729
1C  1170 DATA 74,74,74,56,233,6,
    133,2,133,253,160,5,6,253,38
    ,254,1754
BE  1180 DATA 136,208,249,24,165
    ,251,101,253,133,251,165,252
    ,101,254,133,252,2928
6E  1190 DATA 165,3,240,14,165,2
    ,133,253,169,0,133,3,133,254
    ,160,3,1830
A2  1200 DATA 208,218,96,206,1,2
    08,32,19,170,173,193,2,10,14
    1,1,212,1890
30  1210 DATA 206,193,2,208,23,1
    62,16,142,193,2,232,142,195,
    2,169,0,1887
88  1220 DATA 141,4,212,141,194,
    2,141,197,2,141,196,2,96,29,
    76,75,1649
89  1230 DATA 67,160,41,169,34,2
    09,251,240,7,200,169,35,209,
    251,208,2,2252
04  1240 DATA 24,96,56,96,169,0,
    141,11,212,169,9,141,12,212,
    141,13,1502
07  1250 DATA 212,169,17,141,11,
    212,169,45,141,8,212,96,169,
    0,141,11,1754
CE  1260 DATA 212,169,9,141,12,2
    12,141,13,212,169,33,141,11,
    212,169,8,1864
DA  1270 DATA 141,8,212,96,169,1
    1,141,12,212,141,13,212,169,
    17,141,11,1706
F8  1280 DATA 212,169,224,141,8,
    212,96,173,184,2,73,3,141,18
    4,2,141,1965
CF  1290 DATA 248,7,96,173,66,3,
    208,67,169,2,141,205,2,141,6
    6,3,1597
88  1300 DATA 141,206,2,13,21,20
    8,141,21,208,173,1,208,141,3
    ,208,169,1864
C0  1310 DATA 197,141,249,7,173,
    16,208,72,41,1,240,5,104,9,3
    ,208,1674
F3  1320 DATA 3,104,41,252,141,1
    6,208,56,173,0,208,233,18,14
    1,2,208,1804
37  1330 DATA 176,8,173,16,208,4
    1,253,141,16,208,96,56,173,2
    ,208,233,2008
BE  1340 DATA 4,141,2,208,201,30
    ,176,242,169,0,141,66,3,141,
    205,2,1731
79  1350 DATA 173,21,208,41,253,
    141,21,208,96,173,66,3,208,7
    4,169,2,1857
E3  1360 DATA 141,204,2,141,66,3
    ,141,206,2,13,21,208,141,21,
    208,173,1691
91  1370 DATA 1,208,141,3,208,16
    9,199,141,249,7,173,16,208,7
    2,41,1,1837
8B  1380 DATA 208,6,104,41,252,1
    84,80,6,104,9,3,141,82,3,141
    ,16,1380
EA  1390 DATA 208,24,173,0,208,1
    05,18,141,2,208,144,158,173,
    16,208,9,1795
06  1400 DATA 2,141,16,208,141,8
    2,3,96,24,169,4,172,82,3,208
    ,22,1373
0E  1410 DATA 109,2,208,144,13,7
    2,173,16,208,9,2,141,16,208,
    141,82,1544
8F  1420 DATA 3,104,141,2,208,96
    ,109,2,208,201,60,144,245,16
    9,0,141,1833
0A  1430 DATA 204,2,141,66,3,141
    ,82,3,173,21,208,41,253,141,
    21,208,1708
50  1440 DATA 96,173,203,2,208,3
    7,206,183,2,208,32,162,4,142
    ,183,2,1843
9D  1450 DATA 56,189,88,4,233,1,
    201,48,176,29,72,160,4,185,8
    8,4,1538
B6  1460 DATA 201,48,208,8,136,1
    6,246,104,140,221,2,96,104,2
    4,105,10,1669
12  1470 DATA 157,88,4,202,16,21
    8,96,157,88,4,96,162,0,142,1
    41,3,1574
5C  1480 DATA 142,142,3,232,142,
    143,3,232,56,189,114,4,253,1
    41,3,201,2000
3F  1490 DATA 48,176,2,105,10,15
    7,114,4,202,16,238,232,142,1
    1,212,142,1811
E1  1500 DATA 13,212,169,9,141,1
    2,212,169,3,141,8,212,169,65
    ,141,11,1687
3B  1510 DATA 212,162,0,142,206,
    2,232,142,10,212,173,114,4,2
    01,48,208,2068
CF  1520 DATA 17,173,115,4,201,4
    8,208,10,173,116,4,201,48,20
    8,3,141,1670
63  1530 DATA 247,2,96,169,0,141
    ,247,2,141,207,2,141,141,3,1
    41,142,1822
E1  1540 DATA 3,169,$,141,143,3,
    162,2,24,189,114,4,125,141,3
    ,201,1429
4C  1550 DATA 58,144,2,233,10,15
    7,114,4,202,16,238,96,162,0,
    142,209,1787
62  1560 DATA 2,142,141,3,232,14
    2,142,3,24,189,35,4,125,141,
    3,201,1529
EE  1570 DATA 58,144,2,233,10,15
    7,35,4,202,16,238,238,200,2,
    173,200,1912
FD  1580 DATA 2,201,55,208,77,14
    1,222,2,32,195,160,169,75,32
    ,207,160,1938
21  1590 DATA 32,6,161,96,169,0,
    141,216,2,141,141,3,141,142,
    3,141,1535
CE  1600 DATA 143,3,169,2,141,14
    4,3,162,5,142,145,3,202,24,1
    89,8,1485
39  1610 DATA 4,125,141,3,201,58
    ,144,2,233,10,157,8,4,202,16
    ,238,1546
4D  1620 DATA 232,189,48,4,221,8
    ,4,240,4,176,7,144,6,232,224
    ,4,1743
EF  1630 DATA 208,239,96,173,52,
    4,105,5,201,58,144,2,233,10,
    141,52,1723
5A  1640 DATA 4,173,51,4,105,2,2
    01,58,144,2,233,10,141,51,4,
    162,1345
44  1650 DATA 2,189,48,4,105,0,2
    01,58,144,2,233,10,157,48,4,
    202,1407
4C  1660 DATA 16,239,96,169,0,14
    1,4,212,169,17,141,4,212,169
    ,6,141,1736
1C  1670 DATA 6,212,169,26,141,5
    ,212,96,206,190,2,16,73,160,
    2,140,1656
6C  1680 DATA 190,2,162,7,189,24
    ,34,48,3,24,144,1,56,42,157,
    24,1107
93  1690 DATA 34,189,208,34,48,3
    ,24,144,1,56,42,157,208,34,1
    89,216,1587
AE  1700 DATA 33,72,41,1,208,3,2
    4,144,1,56,104,106,157,216,3
    3,189,1388
DA  1710 DATA 88,34,72,41,1,208,
    3,24,144,1,56,104,106,157,88
    ,34,1161
E1  1720 DATA 202,16,193,136,208
    ,188,96,173,199,2,208,30,169
    ,198,141,248,2407
E2  1730 DATA 7,160,62,185,128,4
    9,153,192,206,136,16,247,169
    ,1,141,236,2088
7D  1740 DATA 2,160,20,140,172,2
    ,140,199,2,96,206,236,2,208,
    52,169,1806
C8  1750 DATA 4,141,236,2,173,22
    0,2,208,73,172,172,2,48,38,1
    60,59,1710
C7  1760 DATA 162,2,185,128,49,1
    53,131,49,136,202,16,246,152
    ,16,241,206,2074
```

```
7F  1770 DATA 172,2,169,0,141,12
    8,49,141,129,49,141,130,49,2
    38,39,208,1785
58  1780 DATA 238,35,208,96,169,
    13,141,35,208,172,219,2,208,
    245,32,35,2056
FA  1790 DATA 135,32,164,167,169
    ,7,141,39,208,169,62,141,172
    ,2,141,220,1969
9D  1800 DATA 2,96,172,172,2,48,
    19,162,2,185,192,206,153,128
    ,49,136,1724
85  1810 DATA 202,16,246,140,172
    ,2,32,248,108,96,169,0,141,2
    20,2,141,1935
F7  1820 DATA 218,2,141,199,2,14
    1,203,2,96,174,167,2,240,3,7
    6,199,1865
52  1830 DATA 171,173,138,3,208,
    22,169,1,141,21,208,141,16,2
    08,169,60,1849
18  1840 DATA 141,0,208,169,182,
    141,1,208,141,138,3,96,56,17
    3,0,208,1865
3E  1850 DATA 233,1,176,10,72,17
    3,16,208,41,254,141,16,208,1
    04,141,0,1794
E6  1860 DATA 208,206,237,2,208,
    28,238,135,3,169,3,141,237,2
    ,45,135,1997
F2  1870 DATA 3,141,135,3,170,18
    9,173,2,141,248,7,224,0,208,
    3,32,1679
6D  1880 DATA 248,108,173,139,3,
    208,60,162,0,32,203,166,160,
    81,169,67,1979
FA  1890 DATA 209,251,208,4,140,
    139,3,96,160,81,169,82,209,2
    51,240,34,2276
0C  1900 DATA 136,209,251,240,29
    ,169,86,209,251,240,23,200,2
    09,251,240,18,2761
FE  1910 DATA 24,173,1,208,105,2
    ,141,1,208,73,255,74,141,1,2
    12,32,1651
AB  1920 DATA 19,170,96,173,0,20
    8,201,174,208,248,173,235,2,
    141,248,7,2303
D0  1930 DATA 141,137,3,32,128,1
    73,96,173,138,3,208,23,162,1
    ,142,21,1581
62  1940 DATA 208,202,142,16,208
    ,169,32,141,0,208,169,182,14
    1,1,208,141,2168
B9  1950 DATA 138,3,96,24,173,0,
    208,105,1,141,0,208,206,237,
    2,208,1750
CA  1960 DATA 28,238,136,3,169,3
    ,141,237,2,45,136,3,141,136,
    3,170,1591
98  1970 DATA 189,177,2,141,248,
    7,224,0,208,3,32,248,108,173
    ,139,3,1902
AC  1980 DATA 208,161,162,0,32,2
    03,166,160,82,169,75,209,251
    ,208,4,140,2230
E6  1990 DATA 139,3,96,160,80,16
    9,82,209,251,240,19,200,209,
    251,240,14,2362
21  2000 DATA 169,30,209,251,240
    ,8,200,209,251,240,3,76,160,
    171,96,65,2378
3E  2010 DATA 19,3,15,18,5,32,65
    ,2,5,19,20,32,32,65,20,9,361
15  2020 DATA 13,5,32,32,65,1,16
    ,16,12,5,19,65,12,9,22,5,329
B2  2030 DATA 19,32,65,2,15,12,2
    0,19,32,169,62,141,73,15,141
    ,150,967
2F  2040 DATA 15,141,180,15,141,
    213,15,141,245,15,141,117,16
    ,141,229,16,1781
E1  2050 DATA 141,8,17,141,41,17
    ,141,88,17,141,153,17,141,25
    3,17,141,1474
11  2060 DATA 42,18,141,65,18,14
    1,132,18,141,1,19,141,96,19,
    141,221,1354
13  2070 DATA 19,141,236,19,141,
    30,20,141,73,20,141,120,20,1
    41,210,20,1492
B0  2080 DATA 141,224,20,141,57,
    21,141,237,21,141,52,22,141,
    98,22,141,1620
50  2090 DATA 181,22,141,218,22,
    141,251,22,141,155,23,141,21
    5,23,141,245,2082
86  2100 DATA 23,141,26,24,141,1
    15,24,141,141,24,141,249,24,
    141,59,25,1439
5A  2110 DATA 141,6,26,141,117,2
    6,141,214,26,141,53,27,141,1
    27,27,141,1495
6F  2120 DATA 153,27,141,241,27,
    141,11,28,141,136,28,141,202
    ,28,141,213,1799
86  2130 DATA 28,141,18,29,141,1
    79,29,141,189,29,141,246,29,
    141,64,30,1575
2A  2140 DATA 96,160,0,132,2,140
    ,114,3,140,219,2,200,132,3,1
    40,218,1701
B1  2150 DATA 2,56,185,75,4,249,
    2,0,201,48,176,2,105,10,153,
    75,1343
F0  2160 DATA 4,136,16,238,173,7
    6,4,201,48,208,11,173,75,4,2
    01,48,1616
E5  2170 DATA 208,4,141,221,2,96
    ,160,10,140,5,212,200,140,12
    ,212,160,1923
FD  2180 DATA 6,140,6,212,136,14
    0,13,212,96,169,48,205,114,4
    ,208,12,1721
CB  2190 DATA 205,115,4,208,7,17
    3,114,216,73,1,240,2,169,1,1
    41,114,1783
B1  2200 DATA 216,141,115,216,14
    1,116,216,96,0,0,0,0,0,0,0
    ,1257
15  2210 DATA 169,170,141,253,7,
    169,171,141,252,7,169,172,14
    1,93,3,141,2199
B6  2220 DATA 94,3,169,219,141,2
    55,7,173,21,208,41,1,9,128,1
    41,21,1631
0E  2230 DATA 208,169,166,141,14
    ,208,169,49,141,15,208,173,2
    9,208,9,128,2035
B9  2240 DATA 141,29,208,169,11,
    141,41,208,141,42,208,141,43
    ,208,141,44,1916
FA  2250 DATA 208,169,0,141,99,3
    ,141,84,3,141,86,3,141,100,3
    ,141,1463
6D  2260 DATA 101,3,169,4,141,46
    ,208,141,102,3,141,88,3,141,
    90,3,1384
86  2270 DATA 141,105,3,141,106,
    3,32,139,178,32,184,178,96,1
    69,0,141,1648
83  2280 DATA 7,212,141,11,212,1
    69,3,141,8,212,169,10,141,12
    ,212,141,1801
19  2290 DATA 13,212,169,33,141,
    11,212,96,173,99,3,240,4,32,
    128,173,1739
3A  2300 DATA 96,162,4,32,28,177
    ,160,41,32,19,176,169,0,105,
    0,141,1342
CE  2310 DATA 96,3,208,22,172,25
    0,7,192,170,144,5,160,154,14
    0,250,7,1980
6C  2320 DATA 32,132,177,169,0,1
    05,0,141,84,3,162,6,32,28,17
    7,160,1408
7E  2330 DATA 40,32,19,176,169,0
    ,105,0,141,98,3,208,22,172,2
    51,7,1443
30  2340 DATA 192,170,144,5,160,
    169,140,251,7,32,132,177,169
    ,0,105,0,1853
B3  2350 DATA 141,86,3,173,5,208
    ,201,228,144,3,32,139,178,17
    3,7,208,1929
36  2360 DATA 201,228,144,3,32,1
    84,178,32,244,174,32,124,175
    ,32,227,178,2188
20  2370 DATA 96,173,100,3,240,3
    3,238,46,208,169,0,141,11,21
    2,169,5,1844
07  2380 DATA 141,12,212,141,13,
    212,24,173,15,208,105,2,141,
    15,208,141,1763
16  2390 DATA 8,212,169,17,141,1
    1,212,173,91,3,240,3,32,96,1
    75,173,1756
9E  2400 DATA 92,3,240,3,32,205,
    175,173,103,3,240,6,206,103,
    3,76,1663
C9  2410 DATA 197,174,32,155,177
    ,173,226,2,240,21,173,225,2,
    240,16,206,2259
A8  2420 DATA 101,3,208,19,206,1
    02,3,208,14,169,1,141,100,3,
    96,141,1515
79  2430 DATA 101,3,169,4,141,10
    2,3,173,104,3,240,4,206,104,
    3,96,1456
F7  2440 DATA 32,19,178,96,173,9
    1,3,208,71,173,1,208,205,5,2
    08,208,1879
86  2450 DATA 63,173,16,208,41,4
    ,208,56,173,96,3,208,51,173,
    250,7,1730
3F  2460 DATA 201,158,176,44,173
    ,4,208,205,0,208,144,36,141,
    8,208,173,2087
09  2470 DATA 1,208,141,9,208,16
    9,158,141,250,7,173,16,208,4
    1,239,141,2110
53  2480 DATA 16,208,173,21,208,
    9,16,141,21,208,141,91,3,32,
    237,173,1698
6E  2490 DATA 96,104,173,21,208,
    41,239,141,21,208,173,96,3,2
    08,5,160,1897
C0  2500 DATA 154,140,250,7,169,
    0,141,91,3,141,8,208,141,9,2
    08,96,1766
42  2510 DATA 56,173,8,208,233,3
    ,176,16,72,173,16,208,73,16,
    170,41,1642
6F  2520 DATA 16,208,206,138,141
    ,16,208,104,141,8,208,96,173
    ,92,3,208,1966
28  2530 DATA 103,173,1,208,205,
    7,208,208,95,173,16,208,41,8
    ,208,88,1950
A1  2540 DATA 173,98,3,208,83,17
    3,251,7,201,167,144,76,173,6
    ,208,201,2172
AA  2550 DATA 40,144,69,205,0,20
    8,176,64,141,10,208,173,1,20
    8,141,11,1799
EC  2560 DATA 208,160,165,140,25
    1,7,173,16,208,41,223,141,16
    ,208,173,21,2151
34  2570 DATA 208,9,32,141,21,20
    8,141,92,3,32,237,173,96,24,
    173,10,1600
91  2580 DATA 208,105,3,144,16,7
    2,173,16,208,73,32,170,41,32
    ,240,9,1542
B8  2590 DATA 138,141,16,208,104
    ,141,10,208,96,104,173,21,20
    8,41,223,141,1973
79  2600 DATA 21,208,173,98,3,20
    8,5,160,169,140,251,7,169,0,
```

```
     141,10,1763
54   2610 DATA 208,141,11,208,141
     ,92,3,96,30,67,75,76,82,84,8
     6,90,1490
D6   2620 DATA 29,35,34,189,92,3,
     208,64,169,34,209,87,240,7,2
     00,169,1769
12   2630 DATA 35,209,87,208,49,1
     89,1,208,205,1,208,240,41,14
     4,39,173,2037
0A   2640 DATA 16,208,61,56,112,2
     08,29,189,0,208,201,36,144,2
     4,189.1.1682
69   2650 DATA 208,233,8,157,1,20
     8,169,172,224,4,208,5,141,25
     0,7,208,2203
C1   2660 DATA 3,141,251,7,56,96,
     24,96,160,80,169,34,209,87,2
     40,244,1897
24   2670 DATA 200,209,87,240,239
     ,200,209,87,240,234,160,80,1
     69,35,209,87,2685
ED   2680 DATA 240,226,200,209,87
     ,240,221,200,209,87,240,216,
     208,216,206,88,3093
CB   2690 DATA 3,208,25,173,93,3,
     73,1,141,93,3,141,250,7,169,
     8,1391
F9   2700 DATA 141,88,3,56,173,5,
     208,233,8,141,5,208,173,226,
     2,208,1878
15   2710 DATA 19,56,173,4,208,23
     3,1,141,4,208,176,8,173,16,2
     08,73,1701
8E   2720 DATA 4,141,16,208,173,2
     25,2,208,19,24,173,4,208,105
     ,1,141,1652
7F   2730 DATA 4,208,144,8,173,16
     ,208,73,4,141,16,208,96,206,
     90,3,1598
56   2740 DATA 208,25,173,94,3,73
     ,1,141,94,3,141,251,7,169,8,
     141,1532
C6   2750 DATA 90,3,56,173,7,208,
     233,8,141,7,208,173,226,2,20
     8,19,1762
68   2760 DATA 56,173,6,208,233,1
     ,141,6,208,176,8,173,16,208,
     73,8,1694
35   2770 DATA 141,16,208,173,225
     ,2,208,196,24,173,6,208,105,
     1,141,6,1833
DE   2780 DATA 208,144,185,173,16
     ,208,73,8,141,16,208,96,169,
     0,133,87,1865
C3   2790 DATA 169,4,133,88,133,9
     0,169,0,133,92,173,16,208,61
     ,56,112,1637
62   2800 DATA 240,11,24,169,32,1
     01,87,133,87,144,2,230,88,18
     9,0,208,1745
1D   2810 DATA 74,74,74,56,233,3,
     24,101,87,133,87,24,189,1,20
     8,105,1473
62   2820 DATA 2,74,74,74,56,233,
     6,133,89,133,91,160,5,6,91,3
     8,1265
90   2830 DATA 92,136,208,249,24,
     165,87,101,91,133,87,165,88,
     101,92,133,1952
95   2840 DATA 88,165,90,240,14,1
     65,89,133,91,169,0,133,90,13
     3,92,160,1852
C8   2850 DATA 3.208.218.96.162,1
     0,160,80,189,8,176,209,87,24
     0,10,200,2056
F2   2860 DATA 209,87,240,5,202,1
     6,239,56,96,24,96,173,96,3,2
     40,4,1786
7E   2870 DATA 32,126,176,96,173,
     84,3,240,20,173,16,208,41,4,
     240,7,1639
2D   2880 DATA 173,4,208,201,82,1
     76,6,238,5,208,238,5,208,206
     ,105,3,2066
54   2890 DATA 208,41,169,4,141,1
     05,3,173,91,3,240,14,172,250
     ,7,192,1813
F7   2900 DATA 161,144,14,160,158
     ,140,250,7,208,17,172,250,7,
     192,157,176,2213
97   2910 DATA 5,238,250,7,208,5,
     160,154,140,250,7,24,173,226
     ,2,240,2089
D6   2920 DATA 6,173,225,2,208,3,
     96,105,2,141,85,3,56,173,4,2
     08,1490
B2   2930 DATA 237,85,3,176,10,72
     ,173,16,208,73,4,141,16,208,
     104,141,1667
16   2940 DATA 4,208,96,173,98,3,
     240,4,32,205,176,96,173,86,3
     ,240,1837
CD   2950 DATA 20,173,16,208,41,8
     ,208,7,173,6,208,201,20,144,
     6,238,1677
92   2960 DATA 7,208,238,7,208,20
     6,106,3,208,41,169,4,141,106
     ,3,173,1828
B6   2970 DATA 92,3,240,14,172,25
     1,7,192,162,208,14,160,165,1
     40,251,7,2078
BA   2980 DATA 208,17,172,251,7,1
     92,166,240,5,206,251,7,208,5
     ,160,169,2264
2F   2990 DATA 140,251,7,24,173,2
     25,2,240,6,173,226,2,208,3,9
     6,105,1881
B2   3000 DATA 2,141,85,3,24,173,
     6,208,109,85,3,144,10,72,173
     ,16,1254
42   3010 DATA 208,73,8,141,16,20
     8,104,141,6,208,96,169,174,1
     41,5,208,1906
30   3020 DATA 169,80,141,4,208,1
     73,16,208,9,4,141,16,208,173
     ,21,208,1779
7D   3030 DATA 9,4,141,21,208,169
     ,154,141,250,7,169,100,141,1
     03,3,169,1789
B7   3040 DATA 0,141,96,3,141,91,
     3,96,169,174,141,7,208,169,1
     00,141,1680
14   3050 DATA 104,3,169,0,141,92
     ,3,141,98,3,141,6,208,173,16
     ,208,1506
34   3060 DATA 41,247,141,16,208,
     173,21,208,9,8,141,21,208,16
     9,169,141,1921
E1   3070 DATA 251,7,96,173,30,20
     8,41,1,240,17,173,21,208,41,
     127,141,1775
C8   3080 DATA 21,208,169,1,141,9
     9,3,141,203,2,96,173,21,208,
     41,2,1529
C5   3090 DATA 240,70,162,4,173,2
     1,208,61,56,112,240,5,32,73,
     179,144,1780
1E   3100 DATA 7,232,232,224,12,2
     08,237,96,189,56,112,73,253,
     45,21,208,2205
AE   3110 DATA 141,21,208,224,4,2
     08,15,173,21,208,41,239,141,
     21,208,32,1905
72   3120 DATA 139,178,32,164,169
     ,96,224,6,208,14,173,21,208,
     41,223,141,2037
6B   3130 DATA 21,208,32,184,178,
     32,164,169,96,56,189,1,208,2
     33,10,205,1986
56   3140 DATA 3,208,176,106,105,
     20,205,3,208,144,99,173,249,
     7,201,199,2106
00   3150 DATA 240,28,173,16,208,
     61,56,112,208,84,56,173,2,20
     8,233,8,1866
11   3160 DATA 221,0,208,176,73,1
     05,16,221,0,208,144,66,24,96
     ,173,16,1747
FA   3170 DATA 208,61,56,112,208,
     29,169,2,45,16,208,208,49,56
     ,173,2,1602
41   3180 DATA 208,233,8,221,0,20
     8,176,38,105,16,176,224,221,
     0,208,144,2186
3A   3190 DATA 29,176,217,24,169,
     2,45,16,208,240,12,173,2,208
     ,105,8,1634
37   3200 DATA 221,0,208,144,9,17
     6,197,173,2,208,105,8,176,24
     2,56,96,2021
C5   3210 DATA 0,0,0,0,0,0,0,0,0,
     0,0,0,0,0,0,0,0
6B   3220 DATA 76,221,180,173,25,
     208,141,25,208,41,1,208,3,76
     ,49,234,1869
72   3230 DATA 169,48,141,18,208,
     173,17,208,41,127,141,17,208
     ,165,255,240,2176
66   3240 DATA 3,76,34,182,173,24
     8,7,201,208,240,107,206,237,
     2,208,23,2155
13   3250 DATA 169,3,141,237,2,24
     ,173,248,7,105,1,201,204,144
     ,5,32,1696
D0   3260 DATA 248,108,169,200,14
     1,248,7,24,173,0,208,105,1,1
     41,0,208,1981
66   3270 DATA 72,201,98,208,5,16
     9,93,141,1,208,104,201,106,2
     08,52,141,2008
36   3280 DATA 2,208,169,101,141,
     1,208,141,3,208,169,208,141,
     248,7,173,2128
80   3290 DATA 21,208,9,2,141,21,
     208,169,0,141,11,212,141,13,
     212,169,1678
58   3300 DATA 9,141,12,212,169,3
     ,141,8,212,169,65,141,11,212
     ,169,1,1675
11   3310 DATA 141,10,212,76,49,2
     34,24,173,16,208,41,2,208,34
     ,173,2,1603
53   3320 DATA 208,105,4,141,2,20
     8,144,235,173,16,208,9,2,141
     ,16,208,1820
6D   3330 DATA 32,81,181,32,116,1
     67,32,164,169,169,32,141,55,
     5,208,211,1795
3B   3340 DATA 173,2,208,105,4,20
     1,100,176,6,141,2,208,76,49,
     234,206,1891
A5   3350 DATA 104,3,208,191,141,
     11,212,206,105,3,208,183,173
     ,21,208,41,2018
B5   3360 DATA 248,141,21,208,169
     ,32,141,104,3,160,0,169,32,1
     53,208,6,1795
67   3370 DATA 153,231,6,169,13,1
     53,208,218,153,231,218,136,2
     08,237,160,6,2500
E1   3380 DATA 169,13,153,26,218,
     136,16,250,132,255,76,34,182
     ,32,88,37,1817
1B   3390 DATA 169,24,141,23,208,
     141,29,208,169,62,133,3,169,
     50,141,7,1677
55   3400 DATA 208,141,9,208,169,
     179,141,251,7,169,184,141,25
     2,7,169,148,2383
91   3410 DATA 141,6,208,169,196,
     141,8,208,169,7,141,42,208,1
     41,43,208,2036
E6   3420 DATA 169,39,133,2,169,2
     4,133,253,169,5,133,254,169,
     0,133,255,2040
21   3430 DATA 133,10,120,169,48,
     141,18,208,169,211,141,20,3,
     169,179,141,1880
```

```
54  3440 DATA 21,3,173,17,208,41
     ,127,141,17,208,173,26,208,9
     ,1,141,1514
E5  3450 DATA 26,208,173,14,220,
     41,254,141,14,220,88,173,236
     ,2,240,251,2301
DC  3460 DATA 96,169,32,160,8,15
     3,241,5,153,25,6,153,65,6,13
     6,16,1424
30  3470 DATA 244,160,6,169,1,15
     3,26,218,185,178,181,153,26,
     6,136,16,1858
92  3480 DATA 242,160,14,185,185
     ,181,153,221,6,169,1,153,221
     ,218,136,16,2261
6C  3490 DATA 242,160,25,185,200
     ,181,153,40,7,169,1,153,40,2
     19,136,16,1927
1D  3500 DATA 242,160,34,185,226
     ,181,153,115,7,169,1,153,115
     ,219,136,16,2112
E9  3510 DATA 242,160,28,185,5,1
     82,153,199,7,169,1,153,199,2
     19,136,16,2054
C1  3520 DATA 242,96,20,8,5,32,5
     ,14,4,25,5,12,12,32,4,15,531
22  3530 DATA 14,5,32,23,9,12,12
     ,25,25,15,21,32,8,1,22,5,261
AF  3540 DATA 32,18,5,19,3,21,4,
     5,4,32,25,15,21,18,32,19,273
52  3550 DATA 15,14,8,15,16,5,32
     ,25,15,21,32,5,14,10,15,25,2
     67
D8  3560 DATA 5,4,32,20,8,5,32,7
     ,1,13,5,32,13,15,18,5,215
5D  3570 DATA 32,20,8,1,14,13,25
     ,32,23,9,6,5,32,5,14,10,249
71  3580 DATA 15,25,5,4,32,13,5,
     32,23,18,9,20,9,14,7,32,263
BB  3590 DATA 9,20,173,236,2,208
     ,82,165,10,208,81,169,128,14
     1,4,212,1848
82  3600 DATA 169,10,141,1,212,1
     41,6,212,162,17,160,0,140,5,
     212,169,1757
92  3610 DATA 129,141,4,212,165,
     253,133,251,165,254,133,252,
     165,3,73,30,2363
AC  3620 DATA 133,3,145,251,24,1
     65,251,105,40,133,251,165,25
     2,105,0,133,2156
1C  3630 DATA 252,202,16,232,24,
     165,253,105,1,133,253,165,25
     4,105,0,133,2293
CD  3640 DATA 254,198,2,16,4,169
     ,1,133,10,76,49,234,206,104,
     3,208,1667
B5  3650 DATA 248,162,0,142,4,21
     2,232,142,236,2,208,237,0,0,
     0,0,1825
1F  3660 DATA 169,0,133,250,169,
     133,133,251,169,0,133,174,13
     3,193,169,160,2369
DF  3670 DATA 133,175,133,194,16
     9,144,133,252,169,155,133,25
     3,160,0,177,250,2630
01  3680 DATA 145,174,230,250,20
     8,2,230,251,230,174,208,2,23
     0,175,165,250,2924
C7  3690 DATA 197,252,208,234,16
     5,251,197,253,208,228,169,43
     ,133,187,169,156,3050
B5  3700 DATA 133,188,169,3,133,
     183,169,0,133,185,160,0,185,
     251,155,240,2287
16  3710 DATA 6,32,210,255,200,2
     08,245,32,207,255,240,251,20
     1,49,240,4,2635
0E  3720 DATA 201,56,48,230,41,1
     5,133,186,76,64,156,147,17,1
     7,73,78,1538
3E  3730 DATA 80,85,84,32,68,69,
     86,73,67,69,32,78,85,77,66,6
     9,1120
F2  3740 DATA 82,13,17,67,65,83,
     61,49,32,47,32,68,73,83,75,6
     1,908
84  3750 DATA 32,56,32,79,82,32,
     57,58,45,32,0,87,84,54,0,0,7
     30
A2  3760 DATA 0,0,0,0,0,0,0,0,0,
     0,0,255,255,255,255,255,1275
A8  3770 DATA 169,54,133,1,32,23
     4,245,169,55,133,1,0,0,0,0,0
     ,1226
```

Personal File

PROGRAM: PERSONAL FILE

```
56  5 IFPEEK(49152)<>32THENLOAD"
     PERS.MC",8,1
62  7 POKE52,100:POKE56,100:POKE
     25600,0:POKE25601,101:POKE25
     602,0
C3  10 P=1:PRINTCHR$(14),"[BLACK
     ]"
FF  20 A$="@0:++DATA0":FORJ=1TOL
     EN(A$):POKE25600+236+J,ASC(M
     ID$(A$,J,1)):NEXT
47  25 A=25856:FORJ=0TO20STEP2:B
     =INT(A/256):POKE25603+J,A-B*
     256
A1  26 POKE25603+J+1,B:A=A+1000:
     NEXT
18  30 GOTO1000
74  90 GOSUB95:IFZ$="N"THEN1000
66  92 GOTO100
05  95 PRINT"[SA]RE YOU SURE?"
7C  96 GETZ$:IFZ$<>"Y"ANDZ$<>"N"
     THEN96
EF  97 RETURN
7C  100 SYS50093:SYS49299
AB  105 GOSUB150:IFEN=0THENPRINT
     "[DOWN]SAVED AS DATA FILE";P
     EEK(25846)-48:GOSUB950:GOTO1
     000
EF  110 IFEN<>63THEN180
8F  120 PRINTEM$,"OVERWRITE IT":
C8  130 GETA$:IFA$<>"Y"ANDA$<>"N
     "THEN130
D5  135 GOSUB95:IFZ$="N"THEN1000
2B  140 IFA$="Y"THENSYS49348:GOT
     O105
5B  145 GOSUB200:IFEN=0THENPRINT
     "NO MORE FILE SPACE":GOSUB95
     0:GOTO1000
6C  147 POKE25600+246,FI:GOTO100
88  150 OPEN15,8,15:INPUT#15,EN,
     EM$:CLOSE15:RETURN
4F  180 PRINT"ERROR -  ",EN,EM$:
     POKE198,0:WAIT198,1:GOTO1000
A1  200 FI=48
43  202 GOSUB210:IFEN<>0THENRETU
     RN
34  205 FI=FI+1:IFFI>58THENEN=0:
     RETURN
7A  207 GOTO202
0A  210 CLOSE3:OPEN3,8,3,"++DATA
     "+CHR$(FI):GOSUB150:RETURN
06  300 PRINT"[DOWN,SW]HICH FILE
      NUMBER(0-9)":INPUTFI:IFFI<0
     ORFI>9THENPRINT"[UP,SPC5]":P
     RINT"[UP3]";:GOTO300
76  310 POKE25846,FI+48:SYS49320
     :GOSUB150:IFEN=0THENPRINT"OK
     ":GOTO350
7A  320 IFEN=62THENPRINT"[SN]O S
     UCH FILE":GOSUB950:GOTO1000
80  330 PRINTEN,EM$:GOSUB950:GOT
     O1000
12  350 SYS50248:P=PEEK(25602)+1
     :GOSUB950:GOTO1000
BB  500 IFP>9THENPRINT"[SN]O MOR
     E SPACE":GOSUB950:GOTO1000
8B  507 PRINT"[CLR,SP]AGE";P;"[H
     OME]";
65  508 POKE251,0:POKE252,4:POKE
     253,0:POKE254,0:POKE25800,5:
     SYS49510
14  510 POKE19,1:INPUTA$:POKE19,
     0:SYS49535
F2  550 SYS49152:P=P+1
25  555 GOTO1000
9B  600 GOSUB702:IFE=1THEN1000
1F  605 POKE780,P1-1:SYS49916:P=
     P-1:GOTO1000
1C  650 E=0:GOSUB705:IFE=1ORP=10
     THEN500
14  655 POKE780,P1-1:SYS50000:P=
     P+1:GOTO1000
7E  700 GOSUB702:IFE=0THEN715
B7  701 GOTO1000
35  702 E=0:IFPEEK(25602)=0THENE
     =1:RETURN
2B  705 PRINT"[SW]HICH PAGE";:IN
     PUTP1
E4  710 IFP1>P-1ORP1=0THENPRINT"
     NO SUCH PAGE":GOSUB950:E=1:R
     ETURN
44  712 RETURN
7E  715 P1=P1-1:POKE780,P1:PRINT
     "[CLR]":SYS49243
E6  717 POKE251,0:POKE252,4:POKE
     253,0:POKE254,0:SYS49510
AF  718 PRINT"[HOME,SP]AGE";P1+1
     ;"[HOME]";
2F  720 POKE19,1:INPUTA$:POKE19,
     0:SYS49535:POKE780,P1:SYS492
     82:GOTO1000
BC  800 IFPEEK(25602)=0THENPRINT
     "[SN]O DATA TO PRINT":GOSUB9
     50:GOTO1000
4E  810 PRINT"[SS]WITCH ON PRINT
     ER AND PRESS A KEY":GOSUB950
5A  820 OPEN4,4,7:SYS49358:PRINT
     #4:SYS65484:IFST=0THENPRINT"
     OK":GOTO840
4A  830 PRINT"ERROR ",ST:GOSUB95
     0:GOTO1000
C8  840 SYS65511:GOSUB950:GOTO10
     00
AD  950 POKE198,0:WAIT198,1:RETU
     RN
71  990 STOP
67  1000 PRINT"[CLR,RVSON,SPC6,S
     P,SE,SR,SS,SO,SN,SA,SL,SSPC,
     SD,SA,ST,SA,SSPC,SF,SI,SL,SE
     ,SSPC,SS,SE,ST,SSPC,SU,SP,SP
     C9,RVSOFF]";
96  1020 PRINT"[SS]ETUP EACH SCR
     EEN USING ANY KEY EXCEPT
7A  1030 PRINT"[RVSON,SR,SE,ST,S
     U,SR,SN,RVSOFF].  [SU]SE [RV
     SON]>[RVSOFF] TO SET THE STA
     RT OF EACH
BA  1031 PRINT"FIELD AND THEN TH
     E [RVSON,SF]1[RVSOFF] AND [R
     VSON,SF]7[RVSOFF] KEYS CAN
35  1032 PRINT"BE USED TO TAB UP
      AND DOWN.
18  1034 PRINT"[RVSON,SF]3[RVSOF
     F] DELETES A LINE[RIGHT]AND
     [RVSON,SF]5[RVSOFF] INSERTS
     ONE.
04  1040 PRINT"[SU]SE THE [RVSON
     ,SR,SE,ST,SU,SR,SN,RVSOFF] K
     EY TO STORE THE PAGE.
FA  1050 PRINT"[DOWN2,SF]UNCTION
     S FROM THIS MENU ARE:-
```

```
26  1060 PRINT"[DOWN] [RVSON,SF]
    1[RVSOFF] - LOAD DATA, [RVSO
    N,SF]7[RVSOFF] SAVE DATA TO
    DISK
E8  1065 PRINT" [RVSON,SF]2[RVSO
    FF] - PRINT DATA IN MEMORY
6B  1070 PRINT" [RVSON,SF]3[RVSO
    FF] - SET-UP NEXT PAGE
F5  1075 PRINT" [RVSON,SF]4[RVSO
    FF] - DELETE A PAGE, [RVSON,
    SF]6[RVSOFF] INSERT ONE
36  1080 PRINT" [RVSON,SF]5[RVSO
    FF] - AMEND A PAGE
07  1095 PRINT" [RVSON,SF]8[RVSO
    FF] - EXIT
87  1100 PRINT"[DOWN2,SP]RESS TH
    E APPROPRIATE FUNCTION KEY
33  1110 GETA$:IFA$=""ORA$<"[F1]
    "ORA$>"[F8]"THEN1110
B6  1120 A=ASC(A$)-132
2C  1130 ONAGOTO300,500,700,90,8
    00,600,650,990
```

PROGRAM: PERS.MC.LDR

```
29  1 PRINT"[CLR]THIS PROGRAM WI
    LL SAVE THE FILE PERS.MC"
AB  2 PRINT"ONTO YOUR OWN DISK."
00  3 PRINT"ONCE THIS IS DONE TH
    IS LOADER PROGRAM "
9C  4 PRINT"IS NO LONGER REQUIRE
    D."
CE  10 BL=75:LN=50:SA=49152
5B  20 FOR L=0 TO BL:CX=0:FOR D=
    0 TO 15:READ A:CX=CX+A:POKE
    SA+L*16+D,A:NEXT D
77  30 PRINT".";:READ A:IF A><CX
     THENPRINT"ERROR IN LINE";LN
    +(L*10):STOP
86  40 NEXT L
75  50 DATA 32,17,192,165,174,14
    1,0,100,165,175,141,1,100,23
    8,2,100,1743
34  60 DATA 96,173,2,100,24,42,1
    70,173,0,100,157,3,100,133,1
    74,232,1679
E2  70 DATA 173,1,100,157,3,100,
    133,175,169,0,133,251,169,4,
    133,252,1953
BA  80 DATA 169,7,133,254,169,23
    2,133,253,32,60,192,96,160,0
    ,177,251,2318
93  90 DATA 145,174,230,174,208,
    2,230,175,230,251,208,2,230,
    252,165,251,2927
AF  100 DATA 197,253,208,234,165
    ,252,197,254,208,228,96,24,4
    2,170,189,3,2720
D2  110 DATA 100,133,251,232,189
    ,3,100,133,252,24,165,251,10
    5,232,133,253,2556
90  120 DATA 165,252,105,3,133,2
    54,169,0,133,174,169,4,133,1
    75,32,60,1961
02  130 DATA 192,96,24,42,170,18
    9,3,100,133,174,232,189,3,10
    0,133,175,1955
3A  140 DATA 76,40,192,32,177,19
    2,169,0,133,251,169,140,133,
    252,169,251,2376
C9  150 DATA 166,253,164,254,32,
    216,255,96,32,177,192,169,0,
    32,213,255,2506
39  160 DATA 96,169,7,162,240,16
    0,100,32,189,255,169,2,162,8
    ,160,1,1912
2D  170 DATA 32,186,255,96,169,1
    0,162,237,32,181,192,76,150,
    192,173,2,2145
47  180 DATA 100,133,2,169,255,1
    33,253,162,4,32,201,255,230,
    253,198,2,2382
43  190 DATA 166,2,16,1,96,165,2
    53,24,42,170,189,3,100,133,2
    51,232,1843
22  200 DATA 189,3,100,133,252,1
    62,24,134,100,160,0,169,146,
    133,254,32,1991
AB  210 DATA 210,255,177,251,133
    ,101,41,127,32,73,193,133,21
    5,169,146,36,2292
64  220 DATA 101,16,16,197,254,2
    08,7,169,18,133,254,32,210,2
    55,165,215,2250
DF  230 DATA 76,41,193,197,254,2
    40,247,208,240,32,210,255,20
    0,192,40,208,2833
CB  240 DATA 209,169,13,32,210,2
    55,165,251,24,105,40,144,2,2
    30,252,133,2234
41  250 DATA 251,198,100,166,100
    ,16,178,48,147,133,215,41,63
    ,6,215,36,1913
DE  260 DATA 215,16,2,9,128,144,
    4,166,212,208,4,112,2,9,64,9
    6,1391
8B  270 DATA 0,0,0,0,0,0,120,173
    ,20,3,141,201,100,173,21,3,9
    55
2F  280 DATA 141,202,100,169,142
    ,141,20,3,169,193,141,21,3,8
    8,96,120,1749
0B  290 DATA 173,201,100,141,20,
    3,173,202,100,141,21,3,88,96
    ,165,197,1824
9E  300 DATA 201,64,208,6,141,20
    0,100,108,201,100,205,200,10
    0,240,248,141,2463
B9  310 DATA 200,100,201,4,240,2
    1,201,3,208,3,76,45,194,201,
    5,208,1910
A2  320 DATA 3,76,87,194,201,6,2
    08,223,76,182,194,164,254,32
    ,215,193,2308
5D  330 DATA 32,215,193,177,251,
    201,62,208,247,32,6,194,132,
    254,166,253,2623
1D  340 DATA 24,32,240,255,76,15
    1,193,136,48,1,96,166,253,22
    4,0,208,2103
B2  350 DATA 21,134,254,138,168,
    24,32,240,255,169,0,133,251,
    169,4,133,2125
84  360 DATA 252,104,104,76,151,
    193,198,253,160,39,56,165,25
    1,233,40,176,2451
7F  370 DATA 2,198,252,133,251,9
    6,200,192,40,240,1,96,166,25
    3,224,24,2368
E1  380 DATA 208,11,169,0,168,17
    0,133,253,133,254,76,229,193
    ,230,253,160,2640
1E  390 DATA 0,24,165,251,105,40
    ,144,2,230,252,133,251,96,16
    4,254,32,2143
FE  400 DATA 6,194,177,251,201,6
    2,208,247,240,143,162,3,181,
    251,157,220,2703
3B  410 DATA 100,202,16,248,96,1
    62,3,189,220,100,149,251,202
    ,16,248,96,2298
AA  420 DATA 24,105,40,144,1,200
    ,96,32,58,194,56,32,240,255,
    134,253,1864
78  430 DATA 169,0,160,4,224,0,2
    08,7,133,251,132,252,76,122,
    194,202,2134
24  440 DATA 32,80,194,202,16,25
    0,133,251,132,252,32,80,194,
    133,34,132,2147
D1  450 DATA 35,166,253,224,24,2
    40,32,160,39,177,34,145,251,
    136,16,249,2181
59  460 DATA 165,34,164,35,133,2
    51,132,252,32,80,194,133,34,
    132,35,230,2036
75  470 DATA 253,166,253,224,24,
    208,224,160,39,169,32,145,25
    1,136,16,251,2551
51  480 DATA 32,69,194,76,151,19
    3,32,58,194,56,32,240,255,13
    4,253,169,2138
84  490 DATA 7,133,35,133,252,16
    9,152,133,34,169,192,133,251
    ,224,24,240,2281
26  500 DATA 214,160,39,177,34,1
    45,251,136,16,249,165,251,16
    4,252,56,233,2542
42  510 DATA 40,176,1,136,133,25
    1,132,252,56,233,40,176,1,13
    6,133,34,1930
4C  520 DATA 132,35,230,253,165,
    253,201,24,208,215,240,171,1
    68,24,42,170,2531
AB  530 DATA 189,3,100,133,174,2
    32,189,3,100,133,175,232,189
    ,3,100,133,2088
13  540 DATA 251,232,189,3,100,1
    33,252,173,0,100,133,253,173
    ,1,100,133,2226
6B  550 DATA 254,206,2,100,173,2
    ,100,24,42,170,189,3,100,141
    ,0,100,1606
68  560 DATA 232,189,3,100,141,1
    ,100,204,2,100,240,8,173,2,1
    00,240,1835
18  570 DATA 4,32,60,192,96,169,
    0,141,0,100,169,101,141,1,10
    0,96,1402
6C  580 DATA 72,24,42,170,189,3,
    100,133,253,232,189,3,100,13
    3,254,173,2070
9F  590 DATA 1,100,133,175,173,0
    ,100,133,174,24,105,232,141,
    0,100,133,1724
AE  600 DATA 251,173,1,100,105,3
    ,141,1,100,133,252,238,2,100
    ,160,0,1760
D0  610 DATA 198,251,165,251,201
    ,255,208,2,198,252,198,174,1
    65,174,201,255,3148
8B  620 DATA 208,2,198,175,177,1
    74,145,251,165,174,197,253,2
    08,226,165,175,2893
12  630 DATA 197,254,208,220,169
    ,147,32,210,255,104,76,130,1
    92,162,24,189,2569
27  640 DATA 0,100,157,0,140,202
    ,16,247,173,246,100,141,25,1
    40,169,26,1882
80  650 DATA 133,253,169,140,133
    ,254,169,0,133,251,169,101,1
    33,252,160,0,2450
1D  660 DATA 177,251,133,174,200
    ,177,251,197,174,240,22,133,
    174,200,192,254,2949
92  670 DATA 208,243,136,32,37,1
    96,32,60,196,173,1,100,197,2
    52,16,222,2101
1E  680 DATA 96,136,240,7,136,32
    ,37,196,32,60,196,160,0,177,
    251,197,1953
C7  690 DATA 174,208,5,200,192,2
    54,208,245,136,132,175,169,2
    55,160,0,145,2658
58  700 DATA 253,200,165,175,145
    ,253,200,165,174,145,253,32,
    46,196,169,3,2574
4B  710 DATA 133,175,76,230,195,
    132,175,177,251,145,253,136,
    16,249,230,175,2748
BD  720 DATA 24,165,251,101,175,
    144,2,230,252,133,251,96,24,
    165,253,101,2367
99  730 DATA 175,144,2,230,254,1
    33,253,96,162,24,189,0,140,1
    57,0,100,2059
31  740 DATA 202,16,247,173,25,1
    40,141,246,100,169,26,133,25
    1,169,140,133,2311
3D  750 DATA 252,169,0,133,253,1
    69,101,133,254,160,0,177,251
    ,201,255,240,2748
6D  760 DATA 23,145,253,200,192,
    244,208,243,132,175,32,46,19
    6,32,60,196,2377
```

```
29  770 DATA 173,1,100,197,254,16,226,96,132,175,32,48,196,32,60,196,1934
8F  780 DATA 160,1,177,251,133,175,200,177,251,164,175,145,253,136,192,255,2845
99  790 DATA 208,249,230,175,32,60,196,169,2,133,175,32,46,196,76,128,2107
B0  800 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
E3  810 PRINT"[CLR,DOWN2,RVSON]PRESS A KEY WHEN READY TO SAVE"
8C  820 GETK$:IFK$=""THEN820
C5  830 POKE43,0:POKE44,192:POKE45,177:POKE46,196:SAVE "PERS.MC",8,1
```

Letter Writer

PROGRAM: LETTER WRITER

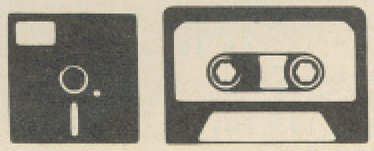
```
B7  1 PRINT"[CLR,C4]"CHR$(8)CHR$(14):POKE650,255:DIMA$(60),TS$(60)
4F  2 T1$=CHR$(176):FORA=1TO62:T1$=T1$+CHR$(192):NEXTA:T1$=T1$+CHR$(174)
D8  3 T2$=CHR$(173):FORA=1TO62:T2$=T2$+CHR$(192):NEXTA:T2$=T2$+CHR$(189)
77  4 Y$="[DOWN24]"
14  5 SP$="[SPC61]"
63  6 L=1:LN=1:TL=1:NA$="[SN]O NAME!":POKE53280,0:POKE53281,0:GOSUB46
40  7 IFPEEK(653)=4THEN64
B4  8 GETK$:IFK$=""THEN7
1C  9 K=ASC(K$)
16  10 IFK=13ANDLN<60THEN25
A7  11 IFK=20THEN27
A0  12 IFK=94THEN39
ED  13 IFK=34THENK$=CHR$(94):GOTO18
7B  14 IFK=147THENA$(LN)="":GOTO171
8A  15 IFK>132ANDK<136THEN161
BE  16 IF(K>31ANDK<94)OR(K>192ANDK<219)ORK=160THEN18
CF  17 GOTO7
DC  18 IFLN=60ANDLEN(A$(60))=60THEN7
6D  19 IFLEN(A$(LN))=59THEN22
7F  20 IFLEN(A$(LN))=60THEN7
89  21 PRINTK$"[CI,LEFT]";:A$(LN)=A$(LN)+K$:GOSUB55:GOTO7
ED  22 PRINTK$;:A$(LN)=A$(LN)+K$:IFLN=60THENPRINT"[CI,LEFT]";:GOTO7
23  23 LN=LN+1:L=L+1:IFL=21THENGOSUB59:GOTO7
7B  24 PRINT:PRINTSPC(10+LEN(A$(LN)))"[CI,LEFT]";:GOSUB55:GOTO7
72  25 PRINT" ":LN=LN+1:L=L+1:IFL=21THENGOSUB59:GOTO7
6A  26 PRINTSPC(10+LEN(A$(LN)))"[CI,LEFT]";:GOSUB55:GOTO7
45  27 IFLN=1ANDLEN(A$(1))=0THEN7
17  28 IFLEN(A$(LN))=0THEN30
15  29 PRINT"[LEFT]  [LEFT2,CI,LEFT]";:A$(LN)=LEFT$(A$(LN),LEN(A$(LN))-1):GOSUB55:GOTO7
27  30 IFL=1ANDLEN(A$(LN))=0THEN34
E6  31 PRINT" ":LN=LN-1:L=L-1
9A  32 IFLEN(A$(LN))>0THENA$(LN)=LEFT$(A$(LN),LEN(A$(LN))-1)
53  33 PRINT"[HOME,DOWN]"LEFT$(Y$,L)SPC(10+LEN(A$(LN)))"[CI,LEFT]";:GOSUB55:GOTO7
BC  34 LN=LN-1:TL=TL-1
F7  35 IFLEN(A$(LN))>0THENA$(LN)=LEFT$(A$(LN),LEN(A$(LN))-1)
BB  36 PRINT"[HOME,DOWN]":FORA=1TO20:PRINTSPC(10)SP$:NEXTA
89  37 PRINT"[HOME,DOWN]":FORA=TLTOTL+19:PRINTSPC(10)A$(A):NEXTA
A5  38 PRINT"[HOME,DOWN2]"SPC(10+LEN(A$(LN)))"[CI,LEFT]";:GOSUB55:GOTO7
37  39 IFLN=1THEN7
6E  40 IFL=1THEN43
35  41 PRINT" ":LN=LN-1:L=L-1:PRINT"[HOME,DOWN]"LEFT$(Y$,L)SPC(10+LEN(A$(LN)))"[CI,LEFT]";
E1  42 GOSUB55:GOTO7
C4  43 PRINT"[HOME,DOWN]":FORA=1TO20:PRINTSPC(10)SP$:NEXTA
F2  44 LN=LN-1:TL=TL-1:PRINT"[HOME,DOWN]":FORA=TLTOTL+19:PRINTSPC(10)A$(A):NEXTA
D6  45 PRINT"[HOME,DOWN2]"SPC(10+LEN(A$(LN)))"[CI,LEFT]";:GOSUB55:GOTO7
7E  46 PRINT"[CLR,SPC5,SL,SE,ST2,SE,SR,SSPC,SW,SR,SI,ST,SE,SR] 2[SPC3]([SC]) 1988[SPC3,SW,SR,SI,ST2,SE,SN,SSPC,SB,SY,SSPC,SA,SD,SA,SM,SSPC,SW,SR,SI,SG,SH,ST]";
E7  47 PRINT" [SF,SO,SR,SSPC,SY,SO,SU,SR,SSPC,SC,SO,SM2,SO,SD,SO,SR,SE]"
01  48 PRINTSPC(8)T1$
A2  49 FORA=1TO20:PRINTSPC(8)"[S-]"SPC(62)"[S-]":NEXTA
E0  50 PRINTSPC(8)T2$:PRINTSPC(22)"[C@36]"
32  51 PRINTSPC(22)"[RVSON]>>> [SP,SR,SE,SS2] "CHR$(34)"[SC,ST,SR,SL]"CHR$(34)" [SF,SO,SR] [SH,SE,SL,SP] [SS,SC,SR,SE2,SN] <<<[HOME]"
8F  52 PRINT"[HOME,DOWN2,SR,SO,SW]:[DOWN,LEFT4,SC,SO,SL]:"
D8  53 GOSUB55
B8  54 RETURN
72  55 PRINT"[HOME,DOWN2,RIGHT4,SPC3,LEFT3]"LNCHR$(13)"[RIGHT4,SPC3,LEFT3]"LEN(A$(LN))+1
96  56 PRINT"[HOME]"LEFT$(Y$,L)
C8  57 PRINTSPC(10+LEN(A$(LN)))"[CI,LEFT]";
B4  58 RETURN
B2  59 IFLN=61THENRETURN
87  60 L=20:PRINT"[HOME,DOWN]":FORA=1TO20:PRINTSPC(10)SP$:NEXTA
27  61 PRINT"[HOME,DOWN]":FORA=TL+1TOTL+20:PRINTSPC(10)A$(A):NEXTA:TL=TL+1
F6  62 PRINT"[HOME,DOWN]"LEFT$(Y$,L)SPC(10+LEN(A$(LN)))"[CI,LEFT]";:GOSUB55
B1  63 RETURN
91  64 PRINT"[CLR]"SPC(26)"[SL,SE,ST2,SE,SR] [SW,SR,SI,ST,SE,SR] 2 [SH,SE,SL,SP] [SS,SC,SR,SE2,SN]"
B6  65 PRINTSPC(26)"[CT28,DOWN]"
01  66 PRINT"[SR,SE,ST,SU,SR,SN] MOVES THE CURSOR TO THE NEXT LINE, AS DOES FILLING A LINE";
DC  67 PRINT" WITH TEXT.[DOWN]"
C4  68 PRINT"[SD,SE,SL,SE,ST,SE] WILL ERASE THE CHARACTER TO THE LEFT OF THE CURSOR.";
8F  69 PRINT" [SI]F THERE IS NO TEXT"
1F  70 PRINT"[SPC7]ON THE CURRENT LINE, IT WILL ERASE THE LAST CHARACTER ON";
E9  71 PRINT" THE LINE ABOVE.[DOWN]"
34  72 PRINT"[SP]RESSING THE 'UP-ARROW' KEY (NEXT TO [SR,SE,SS,ST,SO,SR,SE]) MOVES THE CURSOR";
58  73 PRINT" TO THE END OF THE";
57  74 PRINT"LINE ABOVE, WITHOUT DELETING THE LAST CHARACTER.[DOWN]"
F3  75 PRINT"[ST]YPE YOUR TEXT AS NORMAL. [SA]LL CHARACTERS ARE ACCEPTED, EXCEPT";
53  76 PRINT" THE [SC]OMMODORE"
C4  77 PRINT"KEYBOARD GRAPHICS, THE LEFT-ARROW, AND THE UP-ARROW.[DOWN]"
39  78 PRINT"[SY]OU ARE ALLOWED A MAXIMUM OF SIXTY LINES OF SIXTY CHARACTERS.[DOWN]"
39  79 PRINTSPC(31)"[SY,SO,SU,SR] [SO,SP,ST,SI,SO,SN,SS] [SA,SR,SE]:":PRINTSPC(31)"[CT17]"
C1  80 PRINTSPC(12)"1 = [SR,SE,ST,SU,SR,SN] [ST,SO] [ST,SE,SX,ST,SPC5]4 = [SM,SO,SV,SE] [ST,SE,SX,ST,SPC7]7 = [SS,SA,SV,SE] [ST,SE,SX,ST,DOWN]"
39  81 PRINTSPC(12)"2 = [SG,SO] [ST,SO] [SL,SI,SN,SE,SPC9]5 = [SC,SO,SP,SY] [ST,SE,SX,ST,SPC7]8 = [SL,SO,SA,SD] [ST,SE,SX,ST,DOWN]"
6E  82 PRINTSPC(12)"3 = [SS,SE,ST] [ST,SE,SX,ST] [SN,SA,SM,SE,SPC6]6 = [SD,SE,SL,SE,ST,SE] [ST,SE,SX,ST,SPC5]9 = [SQ,SU,SI,ST] [SP,SR,SO,SG,DOWN]"
7D  83 PRINTSPC(30)"[SP,SL,SE,SA,SS,SE] [SS,SE,SL,SE,SC,ST] (1-9)[HOME]"
C6  84 POKE198,0:WAIT198,1:GETK$:K=VAL(K$):IFK<1ORK>9THEN84
27  85 ONKGOTO86,89,114,93,93,105,138,129,147
8B  86 GOSUB46:PRINT"[HOME,DOWN]":FORA=TLTOTL+19:PRINTSPC(10)A$(A):NEXTA
34  87 GOSUB55:GOTO7
EE  88 PRINT"[HOME,DOWN]"LEFT$(Y$,L)SPC(10+LEN(A$(LN)))"[CI,LEFT]";:GOSUB55:GOTO7
EF  89 PRINT"[CLR,SG,SO] [ST,SO] [SL,SI,SN,SE]":GOSUB158:IFK1$="N"THEN64
35  90 PRINT"[CLR,SE]NTER LINE NUMBER: ";:GOSUB149
4F  91 LN=N:TL=LN:L=1:IFLN>40THENTL=41:L=LN-40
3B  92 GOTO86
9D  93 F$="[SC,SO,SP,SY] [ST,SE,SX,ST]":F=0
FC  94 IFK=4THENF$="[SM,SO,SV,SE] [ST,SE,SX,ST]":F=1
B4  95 PRINT"[CLR]"F$:GOSUB158:IFK1$="N"THEN64
EC  96 PRINT"[CLR,SE]NTER LINE NUMBER OF START OF BLOCK: ";:GOSUB149:SL=N:PRINT
3B  97 PRINT"[SE]NTER LINE NUMBER OF END OF BLOCK  : ";:GOSUB149:EL=N:PRINT
20  98 PRINT"[SE]NTER NEW START LINE NO. FOR BLOCK : ";:GOSU
```

```
    B149:NS=N:PRINT
24  99 IFEL<SLTHENPRINT"[CLR,SE]
    ND LINE LOWER THAN START LIN
    E. [ST]RY AGAIN."
69  100 IFEL<SLTHENFORA=1TO2000:
    NEXTA:GOTO96
0F  101 IFEL-SL>60-NSTHENPRINT"[
    CLR,SB]LOCK WILL NOT FIT.":F
    ORA=1TO2000:NEXTA:GOTO86
2B  102 FORA=0TOEL-SL:TS$(A)=A$(
    SL+A):IFF=1THENA$(SL+A)=""
46  103 NEXTA:FORA=0TOEL-SL:A$(N
    S+A)=TS$(A):TS$(A)="":NEXTA
2F  104 GOTO86
AC  105 PRINT"[CLR,SD,SE,SL,SE,S
    T,SE] [ST,SE,SX,ST]":GOSUB15
    8:IFK1$="N"THEN64
83  106 PRINT"[CLR,SD]O YOU WISH
     TO DELETE A BLOCK, OR THE E
    NTIRE SHEET? ([SB]/[SS])"
74  107 GETK1$:IFK1$<>"B"ANDK1$<
    >"S"THEN107
2B  108 IFK1$="S"THENRUN
77  109 PRINT"[CLR,SE]NTER START
     LINE OF BLOCK TO CLEAR: ";:
    GOSUB149:SL=N:PRINT
FF  110 PRINT"[SE]NTER END LINE
    OF BLOCK TO CLEAR  : ";:GOSU
    B149:EL=N
88  111 IFEL<SLTHENPRINT"[CLR,SE
    ]ND LINE LOWER THAN START LI
    NE. [ST]RY AGAIN."
4C  112 IFEL<SLTHENFORA=1TO2000:
    NEXTA:GOTO109
8C  113 FORA=SLTOEL:A$(A)="":NEX
    TA:GOTO86
95  114 PRINT"[CLR,SS,SE,ST] [ST
    ,SE,SX,ST] [SN,SA,SM,SE,HOME
    ,DOWN2,SC]URRENT NAME: "NA$:
    GOSUB158:IFK1$="N"THEN64
9E  115 PRINT"[CLR,SC]URRENT NAM
    E: "NA$
09  116 PRINT"[DOWN,SE]NTER FILE
    NAME FOR TEXT. ([SM]AXIMUM 1
    6 CHARACTERS)"
E7  117 PRINT"[DOWN]> ..........
    ...... <"CHR$(13)"[UP,RIGHT2
    ]";
87  118 POKE198,0:WAIT198,1:GETK
    $:K=ASC(K$)
8A  119 IFK=13THEN125
01  120 IFK=20THEN127
59  121 IFK>31ANDK<94THEN123
4D  122 GOTO118
B9  123 IFLEN(NM$)=16THEN118
F9  124 PRINTK$;:NM$=NM$+K$:GOTO
    118
67  125 NA$=NM$:NM$="":IFNA$=""T
    HENNA$="[SN]O NAME!"
19  126 GOTO86
A0  127 IFLEN(NM$)=0THEN118
E3  128 PRINT"[LEFT].[LEFT]";:NM
    $=LEFT$(NM$,LEN(NM$)-1):GOTO
    118
D7  129 PRINT"[CLR,SL,SO,SA,SD]
    [ST,SE,SX,ST,HOME,DOWN2,SC]U
    RRENT NAME: "NA$:GOSUB158:IF
    K1$="N"THEN64
33  130 PRINT"[CLR,SI]NSERT TAPE
     CONTAINING TEXT FILE, AND P
    OSITION FOR [SL,SO,SA,SD]."
A1  131 PRINT"[DOWN,ST]HEN PRESS
     [SR,SE,ST,SU,SR,SN] TO CONT
    INUE."
59  132 GETK1$:IFK1$<>CHR$(13)TH
    EN132
89  133 PRINT"[CLR,SL]OADING..."
    :FORA=1TO2000:NEXTA
78  134 IFNA$="[SN]O NAME!"THENO
    PEN1,1,0:GOTO136
79  135 OPEN1,1,0,NA$
65  136 FORA=1TO60:INPUT#1,A$(A)
    :NEXTA:CLOSE1
21  137 L=1:LN=1:TL=1:GOTO86
83  138 PRINT"[CLR,SS,SA,SV,SE]
    [ST,SE,SX,ST,HOME,DOWN2,SC]U
    RRENT NAME: "NA$:GOSUB158:IF
    K1$="N"THEN64
B8  139 PRINT"[CLR,SI]NSERT TAPE
     AND POSITION FOR [SS,SA,SV,
    SE]."
86  140 PRINT"[DOWN,ST]HEN PRESS
     [SR,SE,ST,SU,SR,SN] TO CONT
    INUE."
F2  141 GETK1$:IFK1$<>CHR$(13)TH
    EN141
18  142 PRINT"[CLR,SS]AVING...":
    FORA=1TO2000:NEXTA
7A  143 IFNA$="[SN]O NAME!"THENO
    PEN1,1,1:GOTO145
81  144 OPEN1,1,1,NA$
DE  145 FORA=1TO60:PRINT#1,CHR$(
    34)A$(A)CHR$(34):NEXTA:CLOSE
    1
65  146 GOTO86
5B  147 PRINT"[CLR,SQ,SU,SI,ST]
    [SP,SR,SO,SG,SR,SA,SM]":GOSU
    B158:IFK1$="N"THEN64
6A  148 CLR:PRINT"[CLR]"CHR$(9)C
    HR$(142):POKE53280,12:POKE53
    281,15:END
4D  149 N$=""
0D  150 POKE198,0:WAIT198,1:GETK
    1$:IFK1$=CHR$(13)ANDLEN(N$)>
    0THEN156
07  151 IFK1$=CHR$(20)THEN154
1D  152 IFK1$<"0"ORK1$>"9"ORLEN(
    N$)=2THEN150
EE  153 N$=N$+K1$:PRINTK1$;:GOTO
    150
BA  154 IFLEN(N$)>0THENPRINT"[LE
    FT] [LEFT]";:N$=LEFT$(N$,LEN
    (N$)-1)
A6  155 GOTO150
D5  156 N=VAL(N$):IFN<1ORN>60THE
    N150
13  157 RETURN
96  158 PRINT"[DOWN,SA]RE YOU SU
    RE? ([SY]/[SN])"
2D  159 POKE198,0:WAIT198,1:GETK
    1$:IFK1$<>"Y"ANDK1$<>"N"THEN
    159
2E  160 RETURN
1A  161 IFLEN(A$(LN))=0THEN7
72  162 IFLEFT$(A$(LN),1)=" "THE
    NA$(LN)=RIGHT$(A$(LN),LEN(A$
    (LN))-1):GOTO162
38  163 IFRIGHT$(A$(LN),1)=" "TH
    ENA$(LN)=LEFT$(A$(LN),LEN(A$
    (LN))-1):GOTO163
EF  164 IFLEN(A$(LN))=60THEN7
14  165 IFK=135THEN171
FE  166 TS$(1)=A$(LN):A$(LN)="":
    IFK=134THEN169
10  167 FORS=1TOINT(60-LEN(TS$(1
    )))/2:A$(LN)=A$(LN)+" ":NEXT
    S
BA  168 A$(LN)=A$(LN)+TS$(1):TS$
    (1)="":GOTO171
F9  169 FORS=1TO60-LEN(TS$(1)):A
    $(LN)=A$(LN)+" ":NEXTS
7C  170 A$(LN)=A$(LN)+TS$(1):TS$
    (1)=""
15  171 PRINT"[HOME,DOWN]"LEFT$(
    Y$,L)SPC(10)SP$"[HOME,DOWN]"
    LEFT$(Y$,L)SPC(10)A$(LN)"[CI
    ,LEFT]";
4B  172 GOSUB55:GOTO7
```

PROGRAM: LETTER PRINTER

```
B7  1 PRINT"[CLR,C5]"CHR$(8)CHR$
    (142):POKE53280,0:POKE53281,
    0:DIMA$(60)
12  2 PRINT"[HOME]INSERT THE TAP
    E CONTAINING YOUR TEXT.":PRI
    NT"[DOWN]THEN PRESS RETURN."
27  3 POKE198,0:WAIT198,1:GETK$:
    IFK$<>CHR$(13)THEN3
46  4 PRINT"[CLR]LOADING TEXT...
    PLEASE WAIT.":FORA=1TO2000
    :NEXTA
17  5 OPEN1,1,0:FORA=1TO60:INPUT
    #1,A$(A):NEXTA:CLOSE1
92  6 PRINT"[CLR]PREPARING TEXT
    FOR PRINTING.":PRINT"[DOWN]P
    LEASE WAIT."
7E  7 FORA=1TO60:A$(0)=A$(A):A$(
    A)=""
7E  8 FORB=1TOLEN(A$(0)):B$=MID$
    (A$(0),B,1)
C6  9 IFB$=CHR$(92)THENA$(A)=A$(
    A)+"#":GOTO13
89  10 IFB$=CHR$(94)THENA$(A)=A$
    (A)+CHR$(34):GOTO13
BA  11 IFB$="#"THENA$(A)=A$(A)+C
    HR$(27)+"[SR]"+CHR$(0)+B$+CH
    R$(27)+"[SR]"+CHR$(3):GOTO13
94  12 A$(A)=A$(A)+B$
3E  13 NEXTB,A
60  14 PRINT"[CLR]PRINTING TEXT.
    .."
C1  15 OPEN1,4,7:PRINT#1,CHR$(27
    )"@"CHR$(27)"8"CHR$(27)"3"CH
    R$(18);
FA  16 FORA=1TO60:PRINT#1,"[SPC1
    0]"A$(A):NEXTA
59  17 PRINT#1:CLOSE1:PRINT"[CLR
    ]FINISHED.":END
```

Basic Workshop



PROGRAM: BAS-WRKSHP.LDR

```
2C  1 PRINT"[CLR,DOWN]THIS PROGR
    AM WILL SAVE A COPY OF"
06  2 PRINT"BASIC WORKSHOP ON YO
    UR OWN DISK OR TAPE"
3D  3 PRINT"ONCE THIS IS DONE YO
    U SHOULD LOAD BASIC"
C4  4 PRINT"WORKSHOP AS DETAILED
     IN THE MAGAZINE."
4E  5 PRINT"THIS LOADER PROGRAM
    WILL NO LONGER BE"
1C  6 PRINT"NEEDED."
62  7 PRINT"IF USING TAPE MAKE S
    URE THAT YOU HAVE"
51  8 PRINT"CHANGED THE ,8,1 TO
    ,1,1 AS DETAILED IN"
A1  9 PRINT"THE REM STATEMENTS."
30  10 BL=127:LN=50:SA=49152
5B  20 FOR L=0 TO BL:CX=0:FOR D=
    0 TO 15:READ A:CX=CX+A:POKE
    SA+L*16+D,A:NEXT D
77  30 PRINT".";:READ A:IF A<>CX
     THENPRINT"ERROR IN LINE";LN
    +(L*10):STOP
86  40 NEXT L
E6  50 DATA 76,0,194,234,234,234
    ,234,234,234,234,234,234,234
    ,32,159,255,3056
65  60 DATA 165,197,240,57,201,4
    ,208,6,238,134,2,76,128,192,
    201,5,2054
34  70 DATA 208,23,162,0,173,134
    ,2,157,0,216,157,0,217,157,0
    ,218,1824
63  80 DATA 157,0,219,232,208,24
    1,76,128,192,201,6,208,6,238
    ,33,208,2353
1A  90 DATA 76,128,192,201,3,208
    ,6,238,32,208,76,128,192,173
    ,141,2,2004
B5  100 DATA 41,1,201,1,208,3,76
    ,13,192,173,141,2,74,74,41,1
```

```
    ,1242
A6  110 DATA 201,1,208,38,165,19
    7,201,10,208,7,162,0,169,3,7
    6,0,1646
86  120 DATA 193,201,28,208,24,1
    62,4,169,5,76,0,193,234,234,
    234,234,2199
CD  130 DATA 162,128,160,0,136,2
    08,253,202,208,250,76,49,234
    ,201,20,208,2495
0B  140 DATA 7,162,10,169,5,76,0
    ,193,201,18,208,7,162,16,169
    ,9,1412
A9  150 DATA 76,0,193,201,14,208
    ,7,162,26,169,4,76,0,193,201
    ,26,1556
AB  160 DATA 208,7,162,31,169,4,
    76,0,193,201,29,208,7,162,36
    ,169,1662
1D  170 DATA 5,76,0,193,201,33,2
    08,7,162,42,169,5,76,0,193,2
    01,1571
88  180 DATA 42,208,7,162,48,169
    ,4,76,0,193,201,38,208,7,162
    ,53,1578
77  190 DATA 169,4,76,0,193,201,
    41,208,7,162,58,169,4,76,0,1
    93,1561
60  200 DATA 201,62,208,31,162,6
    3,169,4,76,0,193,76,49,234,2
    34,234,1996
A4  210 DATA 133,198,160,0,189,6
    3,193,153,119,2,232,200,201,
    42,208,244,2337
65  220 DATA 76,128,192,201,17,2
    08,7,162,68,169,6,76,0,193,2
    01,13,1717
95  230 DATA 208,7,162,75,169,4,
    76,0,193,201,22,208,7,162,80
    ,169,1743
83  240 DATA 4,76,0,193,201,31,2
    08,195,162,85,169,6,76,0,193
    ,65,1664
29  250 DATA 83,67,42,71,79,83,8
    5,66,42,67,76,79,83,69,42,76
    ,1110
29  260 DATA 79,65,68,34,36,34,4
    4,56,42,80,69,69,75,42,71,79
    ,943
91  270 DATA 84,79,42,67,72,82,3
    6,40,42,73,78,80,85,84,42,76
    ,1062
D8  280 DATA 73,83,84,42,79,80,6
    9,78,42,80,79,75,69,42,76,79
    ,1130
02  290 DATA 65,68,42,82,69,84,8
    5,82,78,42,83,65,86,69,42,68
    ,1110
A0  300 DATA 65,84,65,42,86,69,8
    2,73,70,89,42,78,69,87,42,25
    5,1298
E4  310 DATA 0,255,0,255,0,255,0
    ,255,158,66,65,83,73,67,32,8
    7,1651
01  320 DATA 79,82,75,83,72,79,8
    0,32,65,67,84,73,86,69,13,67
    ,1106
4B  330 DATA 79,80,89,82,73,71,7
    2,84,32,40,49,57,56,56,41,32
    ,993
F6  340 DATA 66,82,73,65,78,32,7
    1,82,65,72,65,77,46,42,79,76
    ,1071
9F  350 DATA 68,42,42,169,0,141,
    32,208,141,33,208,162,0,160,
    0,136,1542
68  360 DATA 234,234,234,208,250
    ,202,208,247,32,159,255,165,
    197,240,236,96,3197
31  370 DATA 162,0,189,168,193,3
    2,210,255,232,201,46,208,245
    ,120,162,26,2449
23  380 DATA 160,194,142,20,3,14
    0,21,3,88,96,165,197,201,57,
    208,13,1708
5A  390 DATA 173,141,2,74,41,2,2
    01,2,208,3,76,226,252,173,14
    1,2,1717
99  400 DATA 74,74,41,1,201,1,20
    8,110,165,197,201,43,208,7,1
    62,92,1785
89  410 DATA 169,3,76,0,193,201,
    40,208,96,169,3,133,198,162,
    0,189,1840
F5  420 DATA 222,193,157,119,2,2
    32,201,42,208,245,160,3,200,
    177,43,208,2412
ED  430 DATA 251,152,56,101,43,1
    60,0,145,43,133,45,200,165,4
    4,145,43,1726
AF  440 DATA 133,46,160,0,177,45
    ,133,47,200,177,45,170,5,47,
    240,9,1634
8A  450 DATA 134,46,165,47,133,4
    5,136,240,235,24,165,45,105,
    2,133,45,1700
28  460 DATA 133,47,133,49,165,4
    6,105,0,133,46,133,48,133,50
    ,76,128,1425
BB  470 DATA 192,234,234,234,234
    ,234,76,13,192,173,141,2,74,
    74,41,1,2149
D9  480 DATA 201,1,208,242,165,1
    97,201,54,208,236,162,0,189,
    0,4,157,2225
83  490 DATA 0,200,189,0,5,157,0
    ,201,189,0,6,157,0,202,189,0
    ,1495
12  500 DATA 7,157,0,203,189,0,2
    16,157,0,204,189,0,217,157,0
    ,205,1901
D1  510 DATA 189,0,218,157,0,206
    ,189,0,219,157,0,207,232,224
    ,0,208,2206
C8  520 DATA 203,162,0,189,0,196
    ,157,0,4,189,0,197,157,0,5,1
    89,1648
44  530 DATA 0,198,157,0,6,189,0
    ,199,157,0,7,232,224,0,208,2
    27,1804
20  540 DATA 240,66,32,227,193,1
    62,0,189,0,200,157,0,4,189,0
    ,201,1860
BF  550 DATA 157,0,5,189,0,202,1
    57,0,6,189,0,203,157,0,7,189
    ,1461
71  560 DATA 0,204,157,0,216,189
    ,0,205,157,0,217,189,0,206,1
    57,0,1897
16  570 DATA 218,189,0,207,157,0
    ,219,232,224,0,208,203,76,12
    1,195,234,2483
FB  580 DATA 234,234,234,234,162
    ,0,169,15,157,0,216,157,0,21
    7,157,0,2186
DF  590 DATA 218,157,0,219,232,2
    08,239,173,134,2,133,250,173
    ,32,208,133,2511
71  600 DATA 251,173,33,208,133,
    252,76,18,195,165,250,141,13
    4,2,165,251,2447
7C  610 DATA 141,32,208,165,252,
    141,33,208,76,13,192,0,0,0,0
    ,0,1461
55  620 DATA 0,0,0,0,0,0,0,0,0,0
    ,0,0,0,0,0,255,255
9D  630 DATA 0,255,0,255,0,255,0
    ,255,0,255,0,255,0,255,0,255
    ,2040
27  640 DATA 0,255,0,255,0,255,0
    ,255,0,255,0,255,0,255,0,255
    ,2040
E9  650 DATA 0,255,0,255,0,255,0
    ,255,0,255,0,255,0,255,0,255
    ,2040
33  660 DATA 0,255,0,255,0,255,0
    ,255,0,255,0,255,0,255,0,255
    ,2040
C5  670 DATA 0,255,0,255,0,255,0
    ,255,0,255,0,255,0,255,0,255
    ,2040
EB  680 DATA 0,255,0,255,0,255,0
    ,255,0,255,0,255,0,255,243,6
    6,2094
3F  690 DATA 32,32,32,32,32,32,3
    2,2,1,19,9,3,32,23,15,18,346
03  700 DATA 11,19,8,15,16,32,8,
    5,12,16,19,8,5,5,20,46,245
31  710 DATA 32,32,32,32,32,32,3
    2,32,32,32,32,32,32,32,32,45
    ,525
D8  720 DATA 45,45,45,45,45,45,4
    5,45,45,45,45,45,45,45,45,45
    ,720
54  730 DATA 45,45,45,45,45,45,4
    5,45,32,32,32,32,32,32,32,32
    ,616
53  740 DATA 32,32,32,32,32,32,3
    2,32,32,32,32,32,32,32,32,32
    ,512
E5  750 DATA 32,32,32,32,32,32,3
    2,32,32,32,32,32,32,32,32,32
    ,512
A0  760 DATA 32,32,32,32,32,32,3
    2,32,3,20,18,12,32,43,32,1,4
    17
AF  770 DATA 32,61,32,1,19,3,32,
    32,32,32,32,32,32,32,32,32,4
    68
D9  780 DATA 32,32,32,42,3,15,12
    ,15,21,18,19,42,32,32,32,32,
    411
70  790 DATA 3,20,18,12,32,43,32
    ,2,32,61,32,7,15,19,21,2,351
C4  800 DATA 32,32,32,32,32,32,3
    2,32,32,32,32,32,32,32,32,32
    ,512
C5  810 DATA 32,32,32,32,32,32,3
    2,32,3,20,18,12,32,43,32,3,4
    19
E4  820 DATA 32,61,32,3,12,15,19
    ,5,32,32,32,32,32,32,32,32,4
    35
E0  830 DATA 32,32,6,49,32,61,32
    ,3,21,18,19,15,18,32,32,32,4
    34
E7  840 DATA 3,20,18,12,32,43,32
    ,4,32,61,32,12,15,1,4,34,355
45  850 DATA 36,34,44,56,32,32,3
    2,32,32,32,6,51,32,61,32,20,
    564
63  860 DATA 5,24,20,32,32,32,32
    ,32,3,20,18,12,32,43,32,5,37
    4
30  870 DATA 32,61,32,16,5,5,11,
    32,32,32,32,32,32,32,32,32,4
    50
DD  880 DATA 32,32,6,53,32,61,32
    ,19,3,18,5,5,14,32,32,32,408
50  890 DATA 3,20,18,12,32,43,32
    ,7,32,61,32,7,15,20,15,32,38
    1
60  900 DATA 32,32,32,32,32,32,3
    2,32,32,32,6,55,32,61,32,2,5
    08
9B  910 DATA 15,18,4,5,18,32,32,
    32,3,20,18,12,32,43,32,8,324
1C  920 DATA 32,61,32,3,8,18,36,
    40,32,32,32,32,32,32,32,32,4
    86
46  930 DATA 32,32,32,32,32,32,3
    2,32,32,32,32,32,32,32,32,32
    ,512
10  940 DATA 3,20,18,12,32,43,32
    ,9,32,61,32,9,14,16,21,20,37
    4
82  950 DATA 32,32,32,32,32,32,3
    2,19,8,9,6,20,12,15,3,11,327
91  960 DATA 32,61,32,16,1,21,19
```

```
     ,5,3,20,18,12,32,43,32,12,35
     9
B4   970 DATA 32.61,32,12,9,19,20
     ,32,32,32,32,32,32,32,32,32,
     473
B0   980 DATA 32,32,32,32,32,32,3
     2,32,32,32,32,32,32,32,32,32
     ,512
4A   990 DATA 3,20,18,12,32,43,32
     ,15,32,61,32,15,16.5,14,32,3
     82
8C   1000 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
56   1010 DATA 32,32,32,32,32,32,
     32,32,3,20,18,12,32,43,32,16
     ,432
96   1020 DATA 32,61,32.16,15,11.
     5,32,32,32,32,32,32,32,32,32
     ,460
FF   1030 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
D8   1040 DATA 3,20,18,12,32,43,3
     2,17,32,61,32,12,15,1,4,32,3
     66
8B   1050 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
57   1060 DATA 32,32,32.32,32,32,
     32,32,3,20,18,12,32,43,32,18
     ,434
9B   1070 DATA 32.61,32,18.5,20,2
     1,18,14,32,32,32,32,32,32,32
     ,445
29   1080 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
11   1090 DATA 3,20,18,12,32,43,3
     2,19,32,61,32,19,1,22,5,32,3
     83
C5   1100 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
74   1110 DATA 32,32,32,32,32,32,
     32,32,3,20,18,12,32,43,32,20
     ,436
99   1120 DATA 32,61,32,4,1,20,1,
     32,32,32,32,32,32,32,32,32,4
     39
5B   1130 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
71   1140 DATA 3,20,18,12,32,43,3
     2.22,32,61,32,22,5,18,9,6,36
     7
F5   1150 DATA 25,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,505
AA   1160 DATA 32,32,32,32,32,32,
     32,32,3,20,18,12,32,43,32,43
     ,459
9A   1170 DATA 32,61,32,15,12,4,3
     2,32,32,32,32,32,32,32,32,32
     ,476
95   1180 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
6B   1190 DATA 3,20,18,12,32,43,3
     2,45,32,61,32,14,5,23,32,32,
     436
21   1200 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
62   1210 DATA 32,32,32,32,32,32,
     32,32,3,20,18,12,32,43,32,30
     ,446
E8   1220 DATA 32,61.32,8,5,12,16
     ,19,8,5,5,20,32,32,32,32,351

C7   1230 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
54   1240 DATA 3,20,18,12,32,43,3
     2,3,2,13,32,43,32,31,32,61,4
     09
87   1250 DATA 32,18,5,19,5,20,32
     ,32,32,32,32,32,32,32,32,32,
     419
E5   1260 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
EF   1270 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
26   1280 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
1E   1290 DATA 40,3,41,32,49,57,5
     6,56,32,2,18,9,1,14,32,7,449

31   1300 DATA 18,1,8,1,13,32,32,
     32,32,32,32,32,32,32,32,32,3
     93
C4   1310 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,512
4D   1320 DATA 32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,0
     ,480
67   1330 REM ** GET READY TO SAV
     E PROG **
B2   1340 PRINT:PRINT"[RVSON]PRES
     S ANY KEY TO SAVE BASIC WORK
     SHOP"
45   1350 GETK$:IFK$=""THEN1350
F3   1360 POKE43,0:POKE44,192:POK
     E45,0:POKE46,200:SAVE"BASIC
     WORKSHOP",8,1
60   1370 REM CHANGE ,8.1 TO ,1,1
      IF USING A CASSETTE
```

Head for Home

PROGRAM: HEAD FOR HOME

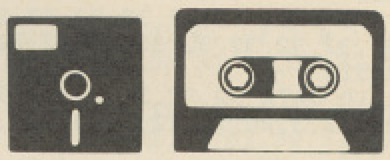
```
10 REM LUDO BY KIRK MCMILLAN
20 POKE65305,119:COLOR1,12,6
30 POKE65301,57:PRINT"[CLEAR][O
14][008][c 4][RVSON]":SYS4452
40 PRINT"[DOWN][RIGHT][RVSOFF][
YELLOW][s W][s I][s R][s E][s C
][s O][s M] [s S][s O][s F][s T
][s W][s A][s R][s E], [s M][s
O][s E][s R][s A][s K][s I], [s
 N][s E][s W] [s Z][s E][s A][s
 L][s A][s N][s D]"
50 COLOR1,16:CHAR,4,20,"[s F][s
 O][s R] [s T][s H][s E] [s C][
s O][s M][s M][s O][s D][s O][s
 R][s E] 16 [s O][s R] [s P][s
L][s U][s S]/4":COLOR1,10
60 CHAR,11,16,"BY [s K]IRK [s M
]C[s M]ILLAN":COLOR1,14:CHAR,7,
18,"[s S]OUND BY [s B]ERNIE [s
M]C[s M]ILLAN"
70 CHAR,15,22,"[WHITE][s L][s O
][s A][s D][s I][s N][s G]":POK
E812,0
80 F=RND(-TI):POKE45,55:POKE46,
54:CLR
90 DLOAD"L.BASIC"
```

PROGRAM: L.BASIC

```
5 SYS 1015:CLR
10 FORF=819TO887:READM$:POKEF,D
EC(M$)
20 NEXT:DATA18,A5,D0,65,D1,65,D
2,65,D3
30 DATA85,D5,60,A2,00,86,03,B5,
D0,C9,29,10,01,60,E8,E0,04,D0,F
4,A2,01,86,03,60
40 DATAA2,00,A9,68,9D,90,09,A9,
A0,9D,90
50 DATA0D,E8,E0,C8,D0,F1,60,A2,
00,A9,68
60 DATA9D,58,0A,A9,A0,9D,58,0E,
E8,E0,C8,D0,F1,60
70 AA=0:AB=0:AC=0:AD=0:AE=0:AF=
0:AG=0:H=0:I=0:M=0:V=0:P=0:N=0:
R=239:TRAP1630
80 C$(0)="[GREEN]":C$(1)="[RED]
":C$(2)="[PURPLE]":C$(3)="[c 7]
":PL$(0)="FIRST":PL$(1)="SECOND
"
90 PL$(2)="THIRD":PL$(3)="AND L
AST"
100 S$="[c P][c A][DOWN][LEFT][
LEFT][c D][s *][UP]"
110 A$(0)="[RVSON][c P][RVSOFF]
[s A][DOWN][LEFT][LEFT][RVSON][
c D][RVSOFF][s B][UP]"
120 A$(1)="[s C][s D][DOWN][LEF
T][LEFT][s E][s F][UP]"
130 A$(2)="[s C][s D][DOWN][LEF
T][LEFT][s G][s H][UP]"
140 A$(3)="[c R][c W][DOWN][LEF
T][LEFT][RVSON][c D][RVSOFF][c
H][UP]"
150 D$(1)="   [DOWN][LEFT][LEFT
][LEFT] [s Q] [DOWN][LEFT][LEFT
][LEFT]   "
160 D$(2)="[s Q]  [DOWN][LEFT][
LEFT][LEFT]   [DOWN][LEFT][LEFT
][LEFT]  [s Q]"
170 D$(3)="[s Q]  [DOWN][LEFT][
LEFT][LEFT] [s Q] [DOWN][LEFT][
LEFT][LEFT]  [s Q]"
180 D$(4)="[s Q] [s Q][DOWN][LE
FT][LEFT][LEFT]   [DOWN][LEFT][
LEFT][LEFT][s Q] [s Q]"
190 D$(5)="[s Q] [s Q][DOWN][LE
FT][LEFT][LEFT] [s Q] [DOWN][LE
FT][LEFT][LEFT][s Q] [s Q]"
200 D$(6)="[s Q] [s Q][DOWN][LE
FT][LEFT][LEFT][s Q] [s Q][DOWN
][LEFT][LEFT][LEFT][s Q] [s Q]"
210 DIMA%(3,3),B%(3,3),C%(3,3),
D%(3,3),M%(3,3),S%(3),E%(3,3),F
%(3,3)
220 DIMG%(3,3),H%(3,3),I%(3,3),
J%(3,3),K%(3,3),L%(3,3),N%(3,3)
230 VOL8:IFPEEK(812)=255THEN330
240 FORT=1375TO1382:POKET,0:NEX
T
250 SOUND1,850,15:CHAR,3,22,"[1
32][s W]OULD YOU LIKE INSTRUCTI
ONS? [YELLOW][130][s Y]/[s N][1
32]":POKER,0
260 GETN$:IFN$<>"Y"ANDN$<>"N"TH
EN260
270 COLOR1,10:SYS1585:PRINT"[CL
EAR][008]"
280 IFN$="N"THEN340
290 COLOR0,10,4:COLOR4,11,5
300 SYS1552:SYS1646:SYS12770
310 COLOR1,5:CHAR,6,24,"PRESS '
RETURN' TO CONTINUE[HOME]":POKE
R,0
320 GETKEYN$:IFN$<>CHR$(13)THEN
320
330 SCNCLR
340 COLOR4,3,5:COLOR0,2:SYS1552
:SYS1681
350 CHAR,3,6,"[c 2]HOW MANY PLA
YERS PLEASE (2-4) [130][c Q][13
2][LEFT]":POKER,0
360 GETKEYN$:IFN$<"2"ORN$>"4"TH
EN360:ELSEPRINTN$:NU=VAL(N$):U=
4*NU:NU=NU-1
370 PRINT:FORF=0TONU:Z=3408+40*
F
380 PRINT"[c 7][RIGHT][RIGHT][R
IGHT]PLAYER"F+1"IS":POKE3,ZAND2
```

```
55:POKE4,Z/256:POKE5,PEEK(3)
390 POKE6,(Z-1024)/256
400 SYS13200:DO:X$(F)=X$(F)+CHR
$(PEEK(Z)+64):Z=Z+1:LOOPUNTILPE
EK(Z)=32:NEXT
410 FORF=OTONU
420 K(F)=INT(RND(1)*(NU+1)):IFS
%(K(F))=1THEN420:ELSES%(K(F))=1
:NEXT:PRINT"[DOWN]"
430 FORZ=OTONU:N$(Z)=X$(K(Z)):S
%(Z)=O:W%(Z)=8:NEXT:FORF=OTONU:
IFF=2THENPRINT"[DOWN]";
440 PRINTC$(F)"  [RVSON][UP]"S$
"[DOWN][RVSOFF] = "LEFT$(N$(F)+
"      ",6),:NEXT:PRINT
450 PRINT"[RIGHT][DOWN][GREEN]M
OVES ARE MADE BY PRESSING THE N
UMBER"
460 PRINT"[RIGHT]MATCHING YOUR
COUNTER, OR 'P' TO PASS":POKER,
O
470 CHAR,9,23,"[RED]PRESS [130]
RETURN[132] TO START"
480 GETN$:IFN$<>CHR$(13)THEN480
:ELSESCNCLR:COLORO,2:COLOR4,9,6
:SYS1716
490 DO:FORL=OTONU
500 POKE208,M%(L,O):POKE209,M%(
L,1)
510 POKE210,M%(L,2):POKE211,M%(
L,3)
520 SYS819:IFPEEK(213)=180THEN1
080
530 CHAR,33,7,"[BLACK][c Q][c Q
][c Q][c Q][c Q][c Q][LEFT][LEF
T][LEFT][LEFT][LEFT][LEFT][RVSO
N]"+N$(L)
540 SOUND1,890,10:CHAR,35,9,C$(
L)+"[RVSON]"+S$+"[RVSOFF]":A=O:
C=A:E=A:D=1
550 DO:COLOR1,12,5:FORF=1TOINT(
RND(1)*15)+15:J=INT(RND(1)*6)+1
560 IFRND(1)<.11THENJ=6
570 SOUND3,850,3
580 IFD>4ORU=1THENJ=INT(RND(1)*
5)+1
590 CHAR,34,17,D$(J):VOL1:SOUND
2,820,2
600 VOL8:NEXT:C=C+J:IFJ<>6THENE
XIT
610 SYS831:E=PEEK(3):IFE=1THENE
XIT
620 D=D+1:CHAR,31+D,22,"[BLACK]
[RVSON]6[RVSOFF]":SOUND1,950,10
:FORF=1TO50:NEXT:SOUND1,950,10:
LOOP
630 PRINTC$(L)
640 IFD>1ORE=1THEN660:ELSEPOKE2
12,C
650 POKE3,1:SYS12211:E=PEEK(3)
660 IFE=1THENSOUND1,516,10:SOUN
D1,262,10:FORF=1TO400:NEXT:GOTO
1080
670 CHAR,33,14,"[130][c Q][132]
[BLACK][c Q][RVSON]":SOUND1,890
,7
680 FORJ=1TOD:CHAR,35,14,"[RVSO
FF][BLACK][c Q][RVSON]":G=O
690 IFC>=6THENA=6:C=C-6:G=1:ELS
EA=C
700 GETKEYA$:IFA$="P"THEN1070:E
LSEIFA$<"1"ORA$>"4"THEN770
710 O=VAL(A$):H=M%(L,O-1)
720 IFH=OANDA<>6ANDG=OTHEN770
730 M=12032:N=22:P=H:IFP>40THEN
M=12178:P=P-40:N=8
740 V=M+N*L+2*(P-1):IFH=OTHEN78
O
750 I=3113+40*PEEK(V+1)+PEEK(V)
760 I=PEEK(I):IFI=66ANDO=1ORI=7
OANDO=2ORI=72ANDO=3ORI=116ANDO=
4THEN780
770 SOUND1,100,10:GOTO700
780 IFH=OANDG=1THENA=1
790 CHAR,35,14,"[RVSON][BLACK]"
+A$:O=O-1
800 IFH>=41ANDH+A<>45THEN770
810 IFH+A>45THEN770
820 M%(L,O)=M%(L,O)+A
830 M$=C$(L)+A$(O):F=FRE(O)
840 POKEA%(L,O),C%(L,O):POKEB%(
L,O),D%(L,O):POKEE%(L,O),G%(L,O
)
850 POKEF%(L,O),D%(L,O):POKEH%(
L,O),J%(L,O):POKEI%(L,O),D%(L,O
)
860 POKEK%(L,O),N%(L,O):POKEL%(
L,O),D%(L,O):COLOR1,9,6:V=12216
+8*L+2*O
870 CHAR,PEEK(V),PEEK(V+1),"[RV
SOFF][c Q][c Q][DOWN][LEFT][LEF
T][c Q][c Q]"
880 FORF=H+1TOM%(L,O):IFF>44THE
N1550
890 M=12032:P=F-1:N=22:IFP>39TH
ENM=12178:P=P-40:N=8
900 V=M+N*L+2*P
910 AA=3072+40*PEEK(V+1)+PEEK(V
)
920 AB=AA-1024:AC=PEEK(AA):AD=P
EEK(AB)
930 AE=PEEK(AA+1):AF=PEEK(AA+40
):AG=PEEK(AA+41):CHAR,PEEK(V),P
EEK(V+1),M$
940 SOUND1,750,5:FORT=1TO100:NE
XT
950 IFF=M%(L,O)THEN980
960 POKEAA,AC:POKEAB,AD:POKEAA+
1,AE:POKEAB+1,AD:POKEAA+40,AF:P
OKEAB+40,AD
970 POKEAA+41,AG:POKEAB+41,AD
980 NEXTF
990 A%(L,O)=AA:B%(L,O)=AB
1000 C%(L,O)=AC:D%(L,O)=AD
1010 E%(L,O)=AA+1:F%(L,O)=AB+1
1020 G%(L,O)=AE:J%(L,O)=AF:N%(L
,O)=AG
1030 H%(L,O)=AA+40:I%(L,O)=AB+4
O
1040 K%(L,O)=AA+41:L%(L,O)=AB+4
1
1050 IFAG=66ORAG=70ORAG=72ORAG=
116THENGOSUB1330
1060 POKE2960+J,16:POKE3984+J,1
07:NEXT
1070 CHAR,33,14,"[RVSOFF][BLACK
][c Q][c Q][c Q]":CHAR,32,22,"[
c Q][c Q][c Q][c Q][c Q][c Q]"
1080 POKER,O:NEXTL
1090 K=-1:FORT=OTONU:K=K-(W%(T)
<4)
1100 NEXT:IFK<NUTHENLOOP
1110 FORF=5TO12:CHAR,32,F,"[BLA
CK][RVSON]         [RVSOFF]"
1120 NEXT:CHAR,34,6,"[RVSON]GAM
E[DOWN][LEFT][LEFT][LEFT][LEFT]
[130]OVER[132]"
1130 CHAR,33,12,"PRESS[DOWN][LE
FT][LEFT][LEFT][LEFT]ANY[DOWN][
LEFT][LEFT][LEFT]KEY"
1140 POKER,O:GETKEYA$:SCNCLR:PL
$(NU)=PL$(3):FORF=OTONU:G=F*5
1150 PRINTC$(W%(F));:FORT=GTOG+
4
1160 PRINT"[RVSON]
                              [RVS
OFF]";:NEXT:PRINT"[UP][UP][UP][
RIGHT]";A$(F);
1170 B$=PL$(F)+" HOME WAS "+N$(
W%(F))
1180 CHAR,19-LEN(B$)/2,G+1,"[RV
SON]"+B$
1190 CHAR,6,G+3,"[RVSON]YOU WER
E SENT BACK"+STR$(S%(W%(F)))+"
TIMES!":PRINT"[DOWN]":NEXT
1200 IFNU<3THENSYS870:IFNU<2THE
NSYS852
1210 POKE1339,PEEK(2048):CHAR,8
,21,"[130]CONGRATULATIONS "+N$(
W%(O))
1220 FORF=12824TO13194STEP5
1230 A=PEEK(F)+PEEK(F+1)*256
1240 B=PEEK(F+2)+PEEK(F+3)*256
1250 C=PEEK(F+4):SOUND1,A,C:SOU
ND2,B,C
1260 NEXT:KEY1,CHR$(136):KEY2,C
HR$(137)
1270 M$=CHR$(34)
1280 CHAR,3,23,"[132][GREEN]"+M
$+"F1"+M$+" TO PLAY AGAIN   [c
7]"+M$+"F2"+M$+" TO EXIT"
1290 POKER,O:GETKEYA$
1300 ONINSTR("[F7][F2]",A$)+1GO
TO1290,1310,1320
1310 POKE812,255:RUN70
1320 SYS65526
1330 X=N%(L,O):Y=D%(L,O)
1340 IFX=64ORX=97ORX=109ORH>40T
HENRETURN
1350 IFX=66THENX=0:ELSEIFX=70TH
ENX=1:ELSEIFX=72THENX=2:ELSEIFX
=116THENX=3
1360 IFY=53THENY=0:ELSEIFY=50TH
ENY=1:ELSEIFY=68THENY=2:ELSEIFY
=46THENY=3
1370 IFY=LTHENRETURN
1380 FORF=700TO200STEP-100:SOUN
D1,F,5:NEXT:SOUND1,50,20:SOUND1
,959,5
1390 V=12216+8*Y+2*X:S%(Y)=S%(Y
)+1
1400 CHAR,PEEK(V),PEEK(V+1),C$(
Y)+A$(X)
1410 A%(L,O)=A%(Y,X):A%(Y,X)=0
1420 B%(L,O)=B%(Y,X):B%(Y,X)=0
1430 C%(L,O)=C%(Y,X):C%(Y,X)=0
1440 D%(L,O)=D%(Y,X):D%(Y,X)=0
1450 E%(L,O)=E%(Y,X):E%(Y,X)=0
1460 F%(L,O)=F%(Y,X):F%(Y,X)=0
1470 G%(L,O)=G%(Y,X):G%(Y,X)=0
1480 H%(L,O)=H%(Y,X):H%(Y,X)=0
1490 I%(L,O)=I%(Y,X):I%(Y,X)=0
1500 J%(L,O)=J%(Y,X):J%(Y,X)=0
1510 K%(L,O)=K%(Y,X):K%(Y,X)=0
1520 L%(L,O)=L%(Y,X):L%(Y,X)=0
1530 N%(L,O)=N%(Y,X):N%(Y,X)=0
1540 M%(Y,X)=0:GOTO1330
1550 V=12248+8*L+2*O
1555 POKE208,M%(L,O):POKE209,M%
(L,1)
1556 POKE210,M%(L,2):POKE211,M%
(L,3)
1557 SYS819:IFPEEK(213)<180THEN
1590
1560 K=O:DO
1570 IFW%(K)<>8THENK=K+1:ELSEEX
IT
1580 LOOP:W%(K)=L
1590 FORZ=400TO700STEP15:SOUND1
,Z,4
1600 FORT=300TOZSTEPZ:SOUND1,T,
2
1610 NEXTT,Z:U=U-1
1620 SOUND2,850,5:SOUND1,850,5:
SOUND1,900,8:CHAR,PEEK(V),PEEK(
V+1),M$:GOTO980
1630 TRAP1630:RESUME
```

Sprite Library



PROGRAM: GEOM.DISPLAY

```
85   10 REM**********************
     ****
31   20 REM* SPRITE LIRBARY DISPL
     AY *
```

```
18   30 REM*        GEOMETRIC
      *
CB   40 REM************************
     ****
7E   50 POKE55,0:POKE56,40:X=X+1:
     IFX=1THENLOAD"GEOMETRIC",8,1

5A   60 V=53248:PRINT"[CLS][BLU][
     26CD][9CR][REV]F7 TO STOP AN
     IMATION"
2C   70 POKEV+21,15:POKEV+23,10:P
     OKEV+29,12:POKEV+32,1:POKEV+
     33,1:POKEV+39,0
40   80 POKEV+40,0:POKEV+41,0:POK
     EV+42,0:POKEV,70:POKEV+1,150
     :POKEV+2,120:POKEV+3,150
5A   90 POKEV+4,170:POKEV+5,150:P
     OKEV+6,248:POKEV+7,150
8D   100 INPUT"[HOM][CD]START SPR
     ITE";S:INPUT"END SPRITE";E:I
     NPUT"DELAY";D
4C   110 FORSP=STOE:FORT=0TOD:NEX
     T:PRINT"[HOM]"TAB(23)"SPRITE
      NO.=";SP:POKE2040,SP
87   120 POKE2041,SP:POKE2042,SP:
     POKE2043,SP:NEXT:GETK$:IFK$=
     "[F7]"THEN100
3D   130 GOTO110
```

PROGRAM: GEOMETRIC DATA

```
AF   10 REM************************
     *
2E   20 REM*   SPRITES LIRBARY
     *
30   30 REM*   ---------------
     *
DA   40 REM*  GEOMETRIC SPRITES
     *
CE   50 REM*  BASIC DATA LOADER
     *
99   60 REM* SPRITES DESIGNED BY
     *
74   70 REM*       MIKE BENN
     *
C5   80 REM************************
     *
DD   90 BL=255  :LN=190  :SA=1024
     0
89   100 FOR L=0 TO BL:CX=0:FOR D
     =0 TO 15
4F   110 READ A:IF A>255THENPRINT
     "NUMBER TO LARGE";LN+(L*10):
     STOP
98   120 CX=CX+A:POKE SA+L*16+D,A
     :NEXT D
D9   130 READ A:IF A><CX THENPRIN
     T"ERROR IN LINE";LN+(L*10):S
     TOP
37   140 NEXTL:POKE43,0:POKE44,40
     :POKE45,0:POKE46,56
84   150 SAVE"GEOMETRIC",8,1:END
EF   160 REM***********************
     ***********
3B   170 REM TAPE USERS WILL NEED
      TO CHANGE           DEVICE N
     UMBER FROM 8 TO 1
FB   180 REM***********************
     ***********
2D   190 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
1B   200 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
01   210 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
71   220 DATA 0,8,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,21,29
75   230 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
63   240 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
79   250 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,8,0,8
72   260 DATA 0,28,0,0,0,0,0,0,0,
     0,0,0,0,0,0,251,279
5C   270 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
4A   280 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
41   290 DATA 0,0,0,0,0,0,0,0,8,0
     ,0,20,0,0,34,0,62
7E   300 DATA 0,65,0,0,128,128,1,
     255,192,0,0,0,0,0,20,789
A4   310 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
92   320 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,8,8
85   330 DATA 0,0,20,0,0,34,0,0,6
     5,0,0,128,128,1,0,64,440
8B   340 DATA 2,0,32,4,0,16,8,0,8
     ,16,0,4,63,255,254,21,683
C8   350 DATA 32,0,0,48,0,0,40,0,
     0,36,0,0,34,0,0,33,223
84   360 DATA 0,0,32,128,0,32,64,
     0,32,32,0,32,16,0,32,8,408
98   370 DATA 0,32,16,0,32,32,0,3
     2,64,0,32,128,0,33,0,0,401
90   380 DATA 34,0,0,36,0,0,40,0,
     0,48,0,0,32,0,0,20,210
09   390 DATA 63,255,254,16,0,4,8
     ,0,8,4,0,16,2,0,32,1,663
E7   400 DATA 0,64,0,128,128,0,65
     ,0,0,34,0,0,20,0,0,8,447
C8   410 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
BE   420 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,132,132
74   430 DATA 0,0,2,0,0,6,0,0,10,
     0,0,18,0,0,34,0,70
C5   440 DATA 0,66,0,0,130,0,1,2,
     0,2,2,0,4,2,0,8,217
AF   450 DATA 2,0,4,2,0,2,2,0,1,2
     ,0,0,130,0,0,66,211
62   460 DATA 0,0,34,0,0,18,0,0,1
     0,0,0,6,0,0,2,0,70
04   470 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
72   480 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,8,8
78   490 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
EE   500 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,242,242
6C   510 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
F3   520 DATA 0,0,0,0,0,0,0,0,0,6
     2,0,0,34,0,0,34,130
13   530 DATA 0,0,34,0,0,62,0,0,0
     ,0,0,0,0,0,0,0,96
13   540 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,131,131
BB   550 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0
EA   560 DATA 0,0,0,255,128,0,128
     ,128,0,128,128,0,128,128,0,1
     28,1279
BE   570 DATA 128,0,128,128,0,128
     ,128,0,128,128,0,255,128,0,0
     ,0,1279
FB   580 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,65,65
F4   590 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,3,255,224,2,484
E0   600 DATA 0,32,2,0,32,2,0,32,
     2,0,32,2,0,32,2,0,170
A9   610 DATA 32,2,0,32,2,0,32,2,
     0,32,2,0,32,2,0,32,202
E9   620 DATA 3,255,224,0,0,0,0,0
     ,0,0,0,0,0,0,0,132,614
D1   630 DATA 0,0,0,0,0,0,15,255,
     248,8,0,8,8,0,8,8,558
09   640 DATA 0,8,8,0,8,8,0,8,8,0
     ,8,8,0,8,8,0,80
07   650 DATA 8,8,0,8,8,0,8,8,0,8
     ,8,0,8,8,0,8,88
4F   660 DATA 8,0,8,8,0,8,15,255,
     248,0,0,0,0,0,0,225,775
D8   670 DATA 0,0,0,0,1,192,0,14,
     64,0,112,64,7,128,64,4,650
DE   680 DATA 0,64,4,0,64,4,0,64,
     4,0,64,4,0,64,4,0,340
A3   690 DATA 64,4,0,64,4,0,64,4,
     0,64,4,0,64,4,0,64,404
AA   700 DATA 3,128,64,0,112,64,0
     ,14,64,0,1,192,0,0,0,93,735
9B   710 DATA 0,0,0,0,3,0,0,5,0,0
     ,9,0,0,17,0,0,34
B2   720 DATA 33,0,0,65,0,1,129,0
     ,1,1,0,1,1,0,1,1,234
90   730 DATA 0,1,1,0,1,1,0,1,129
     ,0,0,65,0,0,33,0,232
BF   740 DATA 0,17,0,0,9,0,0,5,0,
     0,3,0,0,0,0,245,279
CB   750 DATA 0,4,0,0,12,0,0,20,0
     ,0,20,0,0,20,0,0,76
20   760 DATA 36,0,0,36,0,0,36,0,
     0,68,0,0,68,0,0,68,312
8B   770 DATA 0,0,68,0,0,68,0,0,3
     6,0,0,36,0,0,36,0,244
D9   780 DATA 0,20,0,0,20,0,0,20,
     0,0,12,0,0,4,0,14,90
06   790 DATA 0,8,0,0,8,0,0,8,0,0
     ,8,0,0,8,0,0,40
EC   800 DATA 8,0,0,8,0,0,8,0,0,8
     ,0,0,8,0,0,8,48
52   810 DATA 0,0,8,0,0,8,0,0,8,0
     ,0,8,0,0,8,0,40
90   820 DATA 0,8,0,0,8,0,0,8,0,0
     ,8,0,0,8,0,215,255
ED   830 DATA 0,32,0,0,48,0,0,40,
     0,0,40,0,0,40,0,0,200
4B   840 DATA 36,0,0,36,0,0,36,0,
     0,34,0,0,34,0,0,34,210
BD   850 DATA 0,0,34,0,0,34,0,0,3
     6,0,0,36,0,0,36,0,176
89   860 DATA 0,40,0,0,40,0,0,40,
     0,0,48,0,0,32,0,14,214
30   870 DATA 0,0,0,0,96,0,0,80,0
     ,0,72,0,0,68,0,0,316
23   880 DATA 66,0,0,65,0,0,64,19
     2,0,64,64,0,64,64,0,64,707
39   890 DATA 64,0,64,64,0,64,64,
     0,64,192,0,65,0,0,66,0,707
5C   900 DATA 0,68,0,0,72,0,0,80,
     0,0,96,0,0,0,0,53,369
69   910 DATA 0,0,0,3,128,0,2,112
     ,0,2,14,0,2,1,224,2,490
E7   920 DATA 0,32,2,0,32,2,0,32,
     2,0,32,2,0,32,2,0,170
96   930 DATA 32,2,0,32,2,0,32,2,
     0,32,2,0,32,2,0,32,202
B2   940 DATA 2,1,192,2,14,0,2,11
     2,0,3,128,0,0,0,0,62,518
0A   950 DATA 0,0,0,0,0,0,15,255,
     248,8,0,8,8,0,8,8,558
48   960 DATA 0,8,8,0,8,8,0,8,8,0
     ,8,8,0,8,8,0,80
46   970 DATA 8,8,0,8,8,0,8,8,0,8
     ,8,0,8,8,0,8,88
0A   980 DATA 8,0,8,8,0,8,15,255,
     248,0,0,0,0,0,0,22,572
44   990 DATA 0,0,0,0,0,0,0,0,0,0
     ,0,0,31,255,252,16,554
14   1000 DATA 0,4,16,0,4,8,0,8,8
     ,0,8,8,0,8,4,0,76
E2   1010 DATA 16,4,0,16,4,0,16,2
     ,0,32,2,0,32,2,0,32,158
C8   1020 DATA 3,255,224,0,0,0,0,
     0,0,0,0,0,0,0,0,253,735
59   1030 DATA 0,0,0,0,0,0,0,0,0,
     0,0,0,0,0,0,0,0
1A   1040 DATA 0,0,31,255,252,16,
     0,4,8,0,8,4,0,16,2,0,596
6A   1050 DATA 32,1,0,64,0,128,12
     8,0,65,0,0,127,0,0,0,0,545
2B   1060 DATA 0,0,0,0,0,0,0,0,0,
     0,0,0,0,0,0,162,162
A1   1070 DATA 0,0,0,0,0,0,0,0,0,
     0,0,0,0,0,0,0,0
EF   1080 DATA 0,0,0,0,0,0,0,0,63
     ,255,254,16,0,4,14,0,606
DF   1090 DATA 56,1,193,192,0,62,
```

```
    0,0,0,0,0,0,0,0,0,0,504
BF  1100 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,170,170
89  1110 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
2A  1120 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,63,255,318
03  1130 DATA 254,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,254
53  1140 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,156,156
D1  1150 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
98  1160 DATA 0,0,0,0,0,0,0,0,0,
    62,0,1,193,192,14,0,462
AF  1170 DATA 56,16,0,4,63,255,2
    54,0,0,0,0,0,0,0,0,0,648
B9  1180 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,252,252
39  1190 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
43  1200 DATA 0,0,0,255,128,0,12
    8,128,1,0,64,2,0,32,4,0,742
95  1210 DATA 16,8,0,8,16,0,4,32
    ,0,2,63,255,254,0,0,0,658
03  1220 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,145,145
92  1230 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,3,255,224,2,484
4F  1240 DATA 0,32,2,0,32,2,0,32
    ,4,0,16,4,0,16,4,0,144
AF  1250 DATA 16,8,0,8,8,0,8,8,0
    ,8,16,0,4,16,0,4,104
BE  1260 DATA 31,255,252,0,0,0,0
    ,0,0,0,0,0,0,0,0,76,614
37  1270 DATA 0,0,0,0,0,0,15,255
    ,248,8,0,8,8,0,8,8,558
FE  1280 DATA 0,8,8,0,8,8,0,8,8,
    0,8,8,0,8,8,0,80
84  1290 DATA 8,8,0,8,8,0,8,8,0,
    8,8,0,8,8,0,8,88
F2  1300 DATA 8,0,8,8,0,8,15,255
    ,248,0,0,0,0,0,0,223,773
5B  1310 DATA 0,0,0,1,192,0,1,56
    ,0,1,7,0,2,0,224,2,486
6D  1320 DATA 0,28,2,0,4,4,0,4,4
    ,0,8,4,0,8,8,0,74
DD  1330 DATA 8,8,0,16,8,0,16,16
    ,0,16,16,0,32,28,0,32,196
34  1340 DATA 3,128,32,0,112,64,
    0,14,64,0,1,192,0,0,0,174,78
    4
4A  1350 DATA 0,8,0,0,20,0,0,34,
    0,0,65,0,0,128,128,1,384
75  1360 DATA 0,64,2,0,32,4,0,16
    ,8,0,8,16,0,4,32,0,186
03  1370 DATA 2,16,0,4,8,0,8,4,0
    ,16,2,0,32,1,0,64,157
85  1380 DATA 0,128,128,0,65,0,0
    ,34,0,0,20,0,0,8,0,144,527
D0  1390 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,8,0,0,8
47  1400 DATA 54,0,0,65,0,1,128,
    192,6,0,48,24,0,12,32,0,562
E1  1410 DATA 2,24,0,12,6,0,48,1
    ,128,192,0,65,0,0,54,0,532
25  1420 DATA 0,8,0,0,0,0,0,0,0,
    0,0,0,0,0,0,134,142
C8  1430 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
CE  1440 DATA 0,0,0,0,0,0,8,0,0,
    247,128,7,0,112,56,0,558
5C  1450 DATA 14,7,0,112,0,247,1
    28,0,8,0,0,0,0,0,0,0,516
56  1460 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,43,43
10  1470 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
11  1480 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,63,255,318
76  1490 DATA 254,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,254
0A  1500 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,5,5
78  1510 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
BE  1520 DATA 0,0,0,0,0,0,8,0,0,
    247,128,7,0,112,56,0,558
0C  1530 DATA 14,7,0,112,0,247,1
    28,0,8,0,0,0,0,0,0,0,516
51  1540 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,6,6
C7  1550 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,8,0,0,8
88  1560 DATA 54,0,0,65,0,1,128,
    192,6,0,48,24,0,12,32,0,562
E4  1570 DATA 2,24,0,12,6,0,48,1
    ,128,192,0,65,0,0,54,0,532
86  1580 DATA 0,8,0,0,0,0,0,0,0,
    0,0,0,0,0,0,253,261
89  1590 DATA 0,8,0,0,20,0,0,34,
    0,0,65,0,0,128,128,1,384
34  1600 DATA 0,64,2,0,32,4,0,16
    ,8,0,8,16,0,4,32,0,186
76  1610 DATA 2,16,0,4,8,0,8,4,0
    ,16,2,0,32,1,0,64,157
5E  1620 DATA 0,128,128,0,65,0,0
    ,34,0,0,20,0,0,8,0,250,633
41  1630 DATA 0,8,0,0,20,0,0,20,
    0,0,34,0,0,34,0,0,116
49  1640 DATA 65,0,0,65,0,0,128,
    128,1,0,64,1,0,64,2,0,518
87  1650 DATA 32,1,0,64,1,0,64,0
    ,128,128,0,65,0,0,65,0,548
4B  1660 DATA 0,34,0,0,34,0,0,20
    ,0,0,20,0,0,8,0,232,348
65  1670 DATA 0,8,0,0,8,0,0,8,0,
    0,20,0,0,20,0,0,64
DD  1680 DATA 20,0,0,34,0,0,34,0
    ,0,34,0,0,34,0,0,65,221
05  1690 DATA 0,0,34,0,0,34,0,0,
    34,0,0,34,0,0,20,0,156
58  1700 DATA 0,20,0,0,20,0,0,8,
    0,0,8,0,0,8,0,252,316
DB  1710 DATA 0,8,0,0,8,0,0,8,0,
    0,8,0,0,8,0,0,40
71  1720 DATA 8,0,0,8,0,0,8,0,0,
    8,0,0,8,0,0,8,48
D7  1730 DATA 0,0,8,0,0,8,0,0,8,
    0,0,8,0,0,8,0,40
E5  1740 DATA 0,8,0,0,8,0,0,8,0,
    0,8,0,0,8,0,148,188
15  1750 DATA 0,8,0,0,8,0,0,8,0,
    0,20,0,0,20,0,0,64
AD  1760 DATA 20,0,0,34,0,0,34,0
    ,0,34,0,0,34,0,0,65,221
95  1770 DATA 0,0,34,0,0,34,0,0,
    34,0,0,34,0,0,20,0,156
36  1780 DATA 0,20,0,0,20,0,0,8,
    0,0,8,0,0,8,0,19,83
E1  1790 DATA 0,8,0,0,20,0,0,20,
    0,0,34,0,0,34,0,0,116
06  1800 DATA 65,0,0,65,0,0,128,
    128,1,0,64,1,0,64,2,0,518
DC  1810 DATA 32,1,0,64,1,0,64,0
    ,128,128,0,65,0,0,65,0,548
86  1820 DATA 0,34,0,0,34,0,0,20
    ,0,0,20,0,0,8,0,250,366
C8  1830 DATA 0,8,0,0,20,0,0,34,
    0,0,65,0,0,128,128,1,384
D3  1840 DATA 0,64,2,0,32,4,0,16
    ,8,0,8,16,0,4,32,0,186
E5  1850 DATA 2,16,0,4,8,0,8,4,0
    ,16,2,0,32,1,0,64,157
67  1860 DATA 0,128,128,0,65,0,0
    ,34,0,0,20,0,0,8,0,7,390
86  1870 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
FC  1880 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,8,8
FA  1890 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
60  1900 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,101,101
EE  1910 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
BD  1920 DATA 0,0,0,0,0,0,0,0,0,
    8,0,0,20,0,0,34,62
87  1930 DATA 0,0,34,0,0,20,0,0,
    28,0,0,0,0,0,0,0,82
C8  1940 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
36  1950 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
5E  1960 DATA 8,0,0,20,0,0,34,0,
    0,65,0,0,128,128,1,0,384
A2  1970 DATA 64,1,0,64,0,128,12
    8,0,128,128,0,65,0,0,34,0,74
    0
CE  1980 DATA 0,62,0,0,0,0,0,0,0
    ,0,0,0,0,0,0,221,283
8F  1990 DATA 0,0,0,0,0,0,0,8,0,
    0,20,0,0,34,0,0,62
D2  2000 DATA 65,0,0,128,128,1,0
    ,64,2,0,32,4,0,16,8,0,448
DA  2010 DATA 8,8,0,8,4,0,16,4,0
    ,16,2,0,32,1,0,64,163
70  2020 DATA 1,0,64,0,128,128,0
    ,127,0,0,0,0,0,0,0,246,694
60  2030 DATA 0,8,0,0,54,0,0,65,
    0,0,128,128,1,0,64,2,450
4B  2040 DATA 0,32,4,0,16,8,0,8,
    16,0,4,32,0,2,64,0,186
2A  2050 DATA 1,32,0,2,32,0,2,16
    ,0,4,16,0,4,8,0,8,125
A3  2060 DATA 4,0,16,4,0,16,2,0,
    32,2,0,32,1,255,192,216,772
15  2070 DATA 1,255,192,2,0,32,2
    ,0,32,4,0,16,4,0,16,8,564
12  2080 DATA 0,8,16,0,4,16,0,4,
    32,0,2,32,0,2,64,0,180
46  2090 DATA 1,32,0,2,16,0,4,8,
    0,8,4,0,16,2,0,32,125
FD  2100 DATA 1,0,64,0,128,128,0
    ,65,0,0,54,0,0,8,0,34,482
95  2110 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
83  2120 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,8,8
89  2130 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
27  2140 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,30,30
FD  2150 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
0F  2160 DATA 0,0,0,0,0,0,0,0,0,
    8,0,0,54,0,0,193,255
ED  2170 DATA 128,0,54,0,0,8,0,0
    ,0,0,0,0,0,0,0,0,190
CD  2180 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,173,173
C5  2190 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
A2  2200 DATA 0,0,0,28,0,0,99,0,
    1,128,192,2,0,32,2,0,484
91  2210 DATA 32,2,0,32,1,128,19
    2,0,99,0,0,28,0,0,0,0,514
2F  2220 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,5,5
E3  2230 DATA 0,0,0,0,0,0,0,0,0,
    0,28,0,0,99,0,1,128
61  2240 DATA 128,192,6,0,48,8,0
    ,8,8,0,8,8,0,8,8,0,430
AD  2250 DATA 8,8,0,8,8,0,8,8,0,
    8,6,0,48,1,128,192,431
49  2260 DATA 0,99,0,0,28,0,0,0,
    0,0,0,0,0,0,0,21,148
F1  2270 DATA 0,28,0,0,99,0,1,12
    8,192,6,0,48,24,0,12,32,570
8A  2280 DATA 0,2,32,0,2,32,0,2,
    32,0,2,32,0,2,32,0,170
84  2290 DATA 2,32,0,2,32,0,2,32
    ,0,2,32,0,2,32,0,2,172
C7  2300 DATA 24,0,12,6,0,48,1,1
    28,192,0,99,0,0,28,0,117,655
04  2310 DATA 1,255,192,2,0,32,2
    ,0,32,4,0,16,4,0,16,8,564
00  2320 DATA 0,8,8,0,8,16,0,4,1
    6,0,4,32,0,2,32,0,130
30  2330 DATA 2,32,0,2,16,0,4,16
    ,0,4,8,0,8,8,0,8,108
FA  2340 DATA 4,0,16,4,0,16,2,0,
    32,2,0,32,1,255,192,236,792
9D  2350 DATA 0,0,0,0,8,0,0,20,0
    ,0,34,0,0,34,0,0,96
```

```
FC  2360 DATA 65,0,15,128,248,16
    ,0,4,8,0,8,4,0,16,2,0,514
78  2370 DATA 32,4,0,16,8,0,8,16
    ,0,4,15,128,248,0,65,0,544
83  2380 DATA 0,34,0,0,34,0,0,20
    ,0,0,8,0,0,0,0,78,174
05  2390 DATA 1,0,64,1,128,192,1
    ,99,64,1,20,64,1,8,64,1,709
33  2400 DATA 0,64,1,0,64,1,0,64
    ,6,0,48,8,0,8,16,0,280
4C  2410 DATA 4,8,0,8,6,0,48,1,0
    ,64,1,0,64,1,0,64,269
E6  2420 DATA 1,8,64,1,20,64,1,9
    9,64,1,128,128,1,0,64,128,77
    2
4C  2430 DATA 0,28,0,0,99,0,1,12
    8,192,6,0,48,24,0,12,38,576
3C  2440 DATA 0,50,33,128,194,32
    ,99,2,32,28,2,32,8,2,32,8,68
    2
20  2450 DATA 2,32,8,2,32,8,2,32
    ,8,2,32,8,2,32,8,2,212
91  2460 DATA 24,8,12,6,8,48,1,1
    36,192,0,107,0,0,28,0,190,76
    0
3C  2470 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
2A  2480 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,8,8
10  2490 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
92  2500 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,86,86
04  2510 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
11  2520 DATA 0,0,0,0,0,0,0,0,0,
    8,0,0,20,0,0,42,70
E4  2530 DATA 0,0,20,0,0,8,0,0,0
    ,0,0,0,0,0,0,0,28
EA  2540 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,233,233
6C  2550 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
C0  2560 DATA 0,0,0,28,0,0,34,0,
    0,73,0,0,148,128,0,170,581
DE  2570 DATA 128,0,148,128,0,73
    ,0,0,34,0,0,28,0,0,0,0,539
EB  2580 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,43,43
07  2590 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,62,0,0,62
9B  2600 DATA 65,0,0,156,128,1,3
    4,64,2,73,32,2,148,160,2,170
    ,1037
0F  2610 DATA 160,2,148,160,2,73
    ,32,1,34,64,0,156,128,0,65,0
    ,1025
46  2620 DATA 0,62,0,0,0,0,0,0,0
    ,0,0,0,0,0,0,197,259
E6  2630 DATA 0,0,0,0,0,0,0,127,
    0,0,128,128,1,62,64,2,512
15  2640 DATA 65,32,4,156,144,9,
    34,72,10,73,40,10,148,168,10
    ,170,1145
80  2650 DATA 168,10,148,168,10,
    73,40,9,34,72,4,156,144,2,65
    ,32,1135
FB  2660 DATA 1,62,64,0,128,128,
    0,127,0,0,0,0,0,0,0,197,707
90  2670 DATA 0,255,128,1,0,64,2
    ,127,32,4,128,144,9,62,72,18
    ,1046
FD  2680 DATA 65,36,36,156,146,4
    1,34,74,42,73,42,42,148,170,
    42,170,1317
43  2690 DATA 170,42,148,170,42,
    73,42,41,34,74,36,156,146,18
    ,65,36,1293
3C  2700 DATA 9,62,72,4,128,144,
    2,127,32,1,0,64,0,255,128,16
    8,1196
10  2710 DATA 0,255,128,1,0,64,2
    ,0,32,4,0,16,8,0,8,16,534
AB  2720 DATA 0,4,32,0,2,32,0,2,
    32,0,2,32,0,2,32,0,172
CA  2730 DATA 2,32,0,2,32,0,2,32
    ,0,2,32,0,2,16,0,4,158
08  2740 DATA 8,0,8,4,0,16,2,0,3
    2,1,0,64,0,255,0,0,390
```

Electronic Notebook

PROGRAM: ELECTRONIC NOTEBOOK

```
5C  100 REM   AUTHOR L.KEIGHLEY
1D  110 REM   COPYRIGHT 12/11/84

52  120 PRINT"[CLR,DOWN10]"
B7  130 PRINT"[RIGHT14,CI12]"
8F  140 PRINT"[RIGHT14,RVSON] EL
    ECTRONIC "
78  150 PRINT"[RIGHT14,RVSON]  N
    OTEBOOK  "
81  160 PRINT"[RIGHT14,CU12]"
9D  170 X=FRE(0)
A5  180 PRINT"[DOWN2,RIGHT10]COR
    E REQUIRED ";38911-X
39  190 DIMT2$(10),T3$(10),R1(12
    ),D$(11),L$(50)
4E  200 FORX=1TO500:NEXTX:OPEN15
    ,8,15
DE  210 GOTO1640
16  220 PRINTTAB(38);"[S-]"
68  230 RETURN
87  240 F1$="N"
13  250 IFLEN(K$)<LEN(S$)THENRET
    URN
36  260 FORX=1TOLEN(S$)
20  270 IFMID$(S$,X,1)="!"THEN30
    0
6E  280 IFMID$(S$,X,1)=MID$(K$,X
    ,1)THEN300
AD  290 RETURN
07  300 NEXTX
05  310 F1$="Y":RETURN
0F  320 INPUT#15,EN,EM$,ET,ES
FA  330 IFEN=0THENRETURN
8B  340 IFEN=50THENRETURN
3D  350 PRINTCHR$(142)
75  360 PRINT"[CLR]ERROR";E$
86  370 PRINTEM$
46  380 PRINT"[DOWN10]DO YOU WIS
    H TO CONTINUE (Y/N)"
48  390 GETX$:IFX$=""THEN390
3F  400 IFX$<>"N"ANDX$<>"Y"THEN3
    90
B5  410 IFX$="Y"THEN2090
2B  420 RETURN
8E  430 IFE=E1THEN510
35  440 IFE>E1THEN510
90  450 FORX=E+1TOE1
9A  460 RC=X:GOSUB1600:GOSUB320
26  470 IFEN=50THENE=X:GOTO520
36  480 INPUT#1,X$
E5  490 IFLEFT$(X$,1)="[PI]"THEN
    E=X:GOTO520
CF  500 NEXTX
7D  510 E=E+1:E1=E
    R$(S2)+"[PI]"
89  560 X$=X$+STR$(M1)+"[PI]"+P$
    +"[PI]"
09  570 RC=1:GOSUB1600:GOSUB320:
    PRINT#1,X$
C8  580 RETURN
49  590 X1=1:RC=1:GOSUB1600:GOSU
    B320:IFEN=50THENX$="[PI]":GO
    TO610
5F  600 INPUT#1,X$
AB  610 IFLEFT$(X$,1)="[PI]"THEN
    RETURN
0A  620 FORX=1TOLEN(X$)
36  630 IFMID$(X$,X,1)<>"[PI]"TH
    EN780
E3  640 ONX1GOTO650,670,690,710,
    730,750,780
50  650 E=VAL(LEFT$(X$,X-1))
90  660 GOTO760
F3  670 E1=VAL(MID$(X$,X2+1,X-X2
    -1))
9C  680 GOTO760
C5  690 S1=VAL(MID$(X$,X2+1,X-X2
    -1))
88  700 GOTO760
30  710 S2=VAL(MID$(X$,X2+1,X-X2
    -1))
D4  720 GOTO760
EF  730 M1=VAL(MID$(X$,X2+1,X-X2
    -1))
C0  740 GOTO760
05  750 P$=(MID$(X$,X2+1,6))
BE  760 X2=X
7E  770 X1=X1+1
3D  780 NEXTX:X1$="CONTROL"
EF  790 IFX1>6THENRETURN
F0  800 PRINTCHR$(142):PRINT"[CL
    R]ERROR - MISSING FIELD ";X1
    $
AD  810 END
36  820 IFR=E3ANDR=E2THENE2=0:E3
    =0
E0  830 X$=T$+"[PI]"+STR$(S3)+"[
    PI]"+STR$(S4)+"[PI]"+STR$(E2
    )+"[PI]"+STR$(E3)+"[PI]"
3E  840 X$=X$+T1$+"[PI]"+STR$(M2
    )+"[PI]"
AB  850 RC=R:GOSUB1600:GOSUB320:
    PRINT#1,X$
D1  860 RETURN
FD  870 X1=1:RC=R:GOSUB1600:GOSU
    B320:F8=0:IFEN=50THENX$="[PI
    ]":GOTO890
78  880 INPUT#1,X$
BB  890 FORX=1TOLEN(X$)
B1  900 FORX=1TOLEN(X$)
DA  910 IFMID$(X$,X,1)<>"[PI]"TH
    EN1070
54  920 ONX1GOTO930,950,970,990,
    1010,1030,1050,1070
84  930 T$=LEFT$(X$,X-1)
AD  940 GOTO1060
40  950 S3=VAL(MID$(X$,X2+1,X-X2
    -1))
B1  960 GOTO1060
3F  970 S4=VAL(MID$(X$,X2+1,X-X2
    -1))
C5  980 GOTO1060
33  990 E2=VAL(MID$(X$,X2+1,X-X2
    -1))
E9  1000 GOTO1060
86  1010 E3=VAL(MID$(X$,X2+1,X-X
    2-1))
FD  1020 GOTO1060
73  1030 T1$=MID$(X$,X-1,1)
04  1040 GOTO1060
D8  1050 M2=VAL(MID$(X$,X2+1,X-X
    2-1))
1D  1060 X2=X:X1=X1+1
22  1070 NEXTX:X1$="SECTION"
0E  1080 IFX1>6THENRETURN
67  1090 GOTO800
EC  1100 X$=STR$(E4)+"[PI]"+STR$
    (E5)+"[PI]"+K$+"[PI]"+STR$(N
    1)+"[PI]"
88  1110 RC=R2:GOSUB1600:GOSUB32
    0:PRINT#1,X$
EA  1120 RETURN
62  1130 X1=1:RC=R2:GOSUB1600:GO
    SUB320:IFEN=50THENX$="[PI]":
    GOTO1150
7D  1140 INPUT#1,X$
E1  1150 IFLEFT$(X$,1)="[PI]"THE
    NRETURN
E4  1160 FORX=1TOLEN(X$)
C7  1170 IFMID$(X$,X,1)<>"[PI]"T
```

```
    HEN1270
48  1180 ONX1GOTO1190,1210,1230,
    1250,1270
70  1190 E4=VAL(LEFT$(X$,X-1))
A2  1200 GOTO1260
6D  1210 E5=VAL(MID$(X$,X2+1,X-X
    2-1))
46  1220 GOTO1260
A4  1230 K$=MID$(X$,X2+1,X-X2-1)
DA  1240 GOTO1260
DC  1250 N1=VAL(MID$(X$,X2+1,X-X
    2-1))
05  1260 X2=X:X1=X1+1
12  1270 NEXTX:X1$="ENTRY KEY"
ED  1280 IFX1>4THENRETURN
BE  1290 GOTO800
AF  1300 X$=STR$(N2)+"[PI]"+D1$+
    "[PI]"
DE  1310 RC=R3:GOSUB1600:GOSUB32
    0:PRINT#1,X$
A3  1320 RETURN
00  1330 X1=1:RC=R3:GOSUB1600:GO
    SUB320:IFEN=50THENX$="[PI]":
    GOTO1350
16  1340 INPUT#1,X$
16  1350 IFLEFT$(X$,1)="[PI]"THE
    NRETURN
A3  1360 FORX=1TOLEN(X$)
64  1370 IFMID$(X$,X,1)<>"[PI]"T
    HEN1430
7B  1380 ONX1GOTO1390,1410,1430
14  1390 N2=VAL(LEFT$(X$,X-1))
73  1400 GOTO1420
2F  1410 D1$=MID$(X$,X2+1,X-X2-1
    )
28  1420 X2=X:X1=X1+1
90  1430 NEXTX:X1$="ENTRY"
8F  1440 IFX1>2THENRETURN
9E  1450 GOTO800
92  1460 OPEN4,4,1:PRINTCHR$(142
    ):OPEN5,4,7:PRINT#5:CLOSE5:P
    RINTCHR$(14):RETURN
B7  1470 OPEN4,4,7:RETURN
CE  1480 OPEN6,4,8:PRINT#6:CLOSE
    6:CLOSE4:RETURN
2E  1490 RC=2:GOSUB1600:GOSUB320
    :PRINT#1,M:RETURN
80  1500 X1$="":FORX=1TOLEN(D$(1
    0))
BC  1510 IFMID$(D$(10),X,1)="]"T
    HENX1$=X1$+CHR$(44):GOTO1530
95  1520 X1$=X1$+MID$(D$(10),X,1
    )
AE  1530 NEXTX:D$(10)=X1$:RETURN
12  1540 IFE<R2THENRETURN
C0  1550 E=R2:GOTO1580
A1  1560 IFE<R3THENRETURN
80  1570 E=R3
F0  1580 GOSUB540
B2  1590 RETURN
EC  1600 RA=INT(RC/256)
C4  1610 RB=RC-RA*256
DB  1620 PRINT#15,"P"CHR$(98)CHR
    $(RB)CHR$(RA)CHR$(1)
EA  1630 RETURN
52  1640 PRINTCHR$(142):PRINT"[C
    LR,SU,S*37,SI]"
32  1650 PRINT"[S-] 1. INITIALIS
    E NOTEBOOK DATABASE";:GOSUB2
    20:PRINT"[S-]";:GOSUB220
B2  1660 PRINT"[S-] 2. SELECT [R
    VSON]SECTION[RVSOFF] USAGE";
    :GOSUB220
68  1670 PRINT"[S-,SPC4]1. CREAT
    E NEW [RVSON]SECTION[RVSOFF]
    ";:GOSUB220
49  1680 PRINT"[S-,SPC4]2. DELET
    E CURRENT [RVSON]SECTION[RVS
    OFF]";:GOSUB220
40  1690 PRINT"[S-,SPC4]3. AMEND
     CURRENT [RVSON]SECTION[RVSO
    FF]";:GOSUB220
7D  1700 PRINT"[S-,SPC4]4. SELEC
    T [RVSON]SECTION[RVSOFF]";:G
    OSUB220:PRINT"[S-]";:GOSUB22
    0
AA  1710 PRINT"[S-] 3. SELECT [R
    VSON]ENTRIES[RVSOFF] USAGE";
    :GOSUB220
C9  1720 PRINT"[S-,SPC4]1. CREAT
    E NEW [RVSON]ENTRY[RVSOFF]";
    :GOSUB220
C0  1730 PRINT"[S-,SPC4]2. DELET
    E CURRENT [RVSON]ENTRY[RVSOF
    F]";:GOSUB220
E5  1740 PRINT"[S-,SPC4]3. AMEND
     CURRENT [RVSON]ENTRY[RVSOFF
    ]";:GOSUB220
01  1750 PRINT"[S-,SPC4]4. VIEW
    [RVSON]ENTRY[RVSOFF]";:GOSUB
    220
9F  1760 PRINT"[S-,SPC4]5. SELEC
    T [RVSON]ENTRY[RVSOFF]";:GOS
    UB220:PRINT"[S-]";:GOSUB220
00  1770 PRINT"[S-] 4. SELECT [R
    VSON]PRINT[RVSOFF] USAGE";:G
    OSUB220
51  1780 PRINT"[S-,SPC4]1. ALL L
    ABELS";:GOSUB220
D9  1790 PRINT"[S-,SPC4]2. SINGL
    E LABEL";TAB(30);"[SU,S*7,CW
    ]"
34  1800 PRINT"[S-,SPC4]3. ALL R
    EPORT";TAB(30);"[S-] USAGE";
    :GOSUB220
B9  1810 PRINT"[S-,SPC4]4. SINGL
    E REPORT";TAB(30);"[S-]";M;:
    GOSUB220
8E  1820 PRINT"[SJ,S*29,CE,S*7,S
    K]"
D0  1830 PRINT"ENTER REQUIREMENT
     (9 - END RUN)"
5B  1840 GETX$:IFX$=""THEN1840
BB  1850 X=VAL(X$)
A5  1860 IFX<1ORX>9THENPRINT"ERR
    OR - PLEASE RE-ENTER[UP2]":G
    OTO1840
89  1870 ONXGOTO2110,1880,1880,1
    880,1640,1640,1640,1640,2100
F1  1880 IFO1$="Y"THEN1930
CE  1890 OPEN1,8,2,"0:ELECTRICNO
    TEBOOK":O1$="Y":INPUT#15,EN,
    EM$,ET,ES
E3  1900 IFEN<>0THEN2010
4C  1910 GOSUB590
6E  1920 RC=2:GOSUB1600:GOSUB320
    :INPUT#1,M
FF  1930 GETX$:IFX$=""THEN1930
88  1940 X1=VAL(X$)
8D  1950 IFX1=0THEN1840
7F  1960 IFX1>5THEN1930
E0  1970 ONX-1GOTO1980,1990,2000
50  1980 ONX1GOTO2310,2610,2980,
    3550,1930
D3  1990 ONX1GOTO3850,4520,4810,
    6060,6430
D8  2000 ONX1GOTO6980,6980,7410,
    8500,1930
95  2010 IFEN<>62THEN2070
8B  2020 INPUT"ENTER NOTEBOOK NA
    ME ";X$
90  2030 CLOSE1:OPEN1,8,2,"0:ELE
    CTRICNOTEBOOK"
ED  2040 INPUT#15,EN,EM$,ET,ES
86  2050 IFEN<>0THEN2070
18  2060 GOTO1910
C6  2070 PRINT"[CLR]ERROR - CODE
     ";E$;"[DOWN]"
4A  2080 PRINTEM$:CLOSE1:CLOSE15
B2  2090 END
29  2100 PRINT"[CLR]":CLOSE1:CLO
    SE15:GOTO2090
55  2110 PRINT"[CLR]ENTER 'N' BE
    LOW FOR STANDARD NAME[HOME]"
7E  2120 INPUT"[DOWN2]ENTER NOTE
    BOOK NAME ";X$
E8  2130 IFX$="N"ORX$="[SN]"THEN
    2170
F5  2140 CLOSE1
16  2150 OPEN1,8,2,"0:"+X$+",L,"
    +CHR$(42)
7A  2160 GOTO2180
A9  2170 CLOSE1:OPEN1,8,2,"0:ELE
    CTRICNOTEBOOK,L,"+CHR$(42)
51  2180 INPUT"ENTER MAXIMUM NO.
     OF RECORDS ";M
7A  2190 IFM>3500ORM<500THEN2180
12  2200 FORRC=3TOM:PRINT"[CLR]R
    ECORDS SET UP :- ";RC
AE  2210 GOSUB1600
8E  2220 PRINT#1,"[PI]"
CD  2230 NEXTRC
48  2240 E=26:E1=26:S1=0:S2=0:M1
    =0
D1  2250 PRINT"[CLR]ENTER POINTE
    R CHANGE PASSWORD";
EF  2260 I3=6:I4=6:GOSUB9200
95  2270 P$=I$
B6  2280 GOSUB540
28  2290 M=M-2:GOSUB1490
7B  2300 CLOSE1:GOTO1640
5A  2310 PRINTCHR$(14):PRINT"[CL
    R,RVSON,SC,SR,SE,SA,ST,SE,SS
    PC,SS,SE,SC,ST,SI,SO,SN,RVSO
    FF,SSPC]([SSPC,SE,SN,ST,SE,S
    R,SSPC]'[SE,SO,SF]'[SSPC,ST,
    SO,SSPC,SE,SN,SD]"
FD  2320 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN,SSPC,ST,SI,ST,SL,S
    E] ";
52  2330 I3=10:I4=1:GOSUB9200
6F  2340 IFI$="RETURN"ORI$="*E*"
    ORI$="END"ORI$="EOF"THENPRIN
    TCHR$(142):GOTO1640
5A  2350 T$=I$
A7  2360 S3=1:S4=S2:E2=0:E3=0:M2
    =M1+1
2F  2370 PRINT"[DOWN2,SPC5,ST]-[
    ST,SE,SX,ST]":PRINT"[SPC5,SL
    ]-[SL,SA,SB,SE,SL,SS]":PRINT
    "[SPC5,SN]-[SN,SO,ST,SE,SS,U
    PS]"
95  2380 INPUT"[DOWN,SU,SS,SA,SG
    ,SE,SSPC,ST,SY,SP,SE,SSPC]([
    SL]/[SN]/[ST]) ";T1$
48  2390 IFT1$<>"[ST]"ANDT1$<>"[
    SN]"ANDT1$<>"[SL]"THENPRINT"
    [UP3]":GOTO2380
3E  2400 PRINT"[DOWN10]"
93  2410 IFS2<3THENR=3:GOTO2470
A3  2420 FORR=S1TOS2
FD  2430 RC=R:GOSUB1600:GOSUB320
    :INPUT#1,X$
EF  2440 IFT$=LEFT$(X$,LEN(T$))T
    HEN2580
6B  2450 NEXTR
B7  2460 R=S2+1
F2  2470 IFS4<3THENS4=1
8E  2480 GOSUB820
67  2490 S2=R:M1=M1+1
13  2500 IFS1=0THENS1=R
EB  2510 GOSUB540
8F  2520 M=M-1
EC  2530 GOSUB1490
BA  2540 R9=R:R=S4:GOSUB870:S3=R
    9:GOSUB820:R=R9:GOSUB870
14  2550 PRINT"[RIGHT11,RVSON,SR
    ,SE,SC,SO,SR,SD,SSPC,SI,SN,S
    S,SE,SR,ST,SE,SD]"
BB  2560 FORX=1TO1000:NEXTX
14  2570 PRINTCHR$(142):GOTO1640
98  2580 PRINT"[RVSON,SE,SR2,SO,
    SR,RVSOFF,SSPC,C2,SSPC,SD,SU
    ,SP,SL,SI,SC,SA,ST,SE,SSPC,S
    T,SI,ST,SL,SE,SSPC,C2,SSPC,S
    P,SL,SE,SA,SS,SE,SSPC,SR,SE,
    ST,SR,SY]"
```

```
57  2590 FORX=1TO1200:NEXTX
3F  2600 GOTO2310
39  2610 PRINTCHR$(14):PRINT"[CL
    R,SD,SE,SL,SE,ST,SE,SSPC,SC,
    SU,SR2,SE,SN,ST,SSPC,SS,SE,S
    C,ST,SI,SO,SN]"
F1  2620 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN,SSPC]:- ";T$
E8  2630 PRINT"[DOWN2,SA,SR,SE,S
    SPC,SY,SO,SU,SSPC,SS,SU,SR,S
    E,SSPC]([SY]/[SN])"
CF  2640 GETX$:IFX$=""THEN2640
7A  2650 IFX$="N"THENPRINTCHR$(1
    42):GOTO1640
DB  2660 IFX$<>"Y"THEN2640
EA  2670 PRINT"[DOWN2,RIGHT7,RVS
    ON,SD,SE,SL,SE,ST,SI,SN,SG,S
    SPC,SC,SU,SR2,SE,SN,ST,SSPC,
    SS,SE,SC,ST,SI,SO,SN]"
B9  2680 R$="[PI]"
D5  2690 IFR<>0THEN2720
7E  2700 PRINT"[DOWN3,RVSON,SE,S
    R2,SO,SR,RVSOFF] [SN,SO,SSPC
    ,SC,SU,SR2,SE,SN,ST,SSPC,SS,
    SE,SC,ST,SI,SO,SN,SSPC,SR,SE
    ,SC,SO,SR,SD]"
94  2710 GOTO2560
40  2720 RC=R:GOSUB1600:GOSUB320
    :PRINT#1,R$:M=M+1:GOSUB1490
D8  2730 PRINT"[DOWN2,RIGHT11,RV
    SON,SD,SE,SL,SE,ST,SI,SN,SG,
    SSPC,SE,SN,ST,SR,SI,SE,SS]"
7D  2740 IFE2=0THEN2860
F1  2750 R2=E2
E4  2760 GOSUB1130
93  2770 RC=R2:GOSUB1600:PRINT#1
    ,R$:M=M+1:GOSUB1490:GOSUB154
    0
A5  2780 IFN1<>0THENR3=N1:GOTO28
    10
07  2790 IFE4=RTHEN2860
C1  2800 R2=E4:GOTO2760
BC  2810 GOSUB1330
25  2820 RC=R3:GOSUB1600:PRINT#1
    ,R$:M=M+1:GOSUB1490:GOSUB156
    0
C9  2830 IFN2=0THEN2790
56  2840 R3=N2
2F  2850 GOTO2810
A7  2860 R8=S3:R9=S4
A3  2870 IFR8<>1THEN2890
3F  2880 S2=R9:GOSUB540:GOTO2900
B0  2890 R=R8:GOSUB870:S4=R9:GOS
    UB820
BE  2900 IFR9<>1THEN2920
DC  2910 S1=R8:GOSUB540:GOTO2930
E9  2920 R=R9:GOSUB870:S3=R8:GOS
    UB820
41  2930 T$="":S3=0:S4=0:E2=0:E3
    =0:T1$="":M2=0
4D  2940 N2=0:R=0:R1=0:E4=0:E5=0
    :N1=0:K$=""
42  2950 PRINT"[DOWN2,RIGHT11,RV
    SON,SD,SE,SL,SE,ST,SI,SO,SN]
    [SC,SO,SM,SP,SL,SE,ST,SE]"
0D  2960 FORX=1TO750:NEXTX
E7  2970 PRINTCHR$(142):GOTO1640
2C  2980 PRINTCHR$(14):IFZ1$="E"
    ORZ1$="Y"THEN3000
2C  2990 Z1$="N"
E0  3000 PRINT"[CLR,RVSON,SA,SM,
    SE,SN,SD,SSPC,SS,SE,SC,ST,SI
    ,SO,SN,DOWN3]"
42  3010 PRINT"1.  [ST,SI,ST,SL,
    SE] :-[RVSON,LEFT]";T$
B2  3020 PRINT"2.  [ST,SY,SP,SE]
    :-[RVSON,LEFT]";T1$
1D  3030 IFZ1$<>"Y"THEN3090
D7  3040 PRINT"3.  [SN,SE,SX,ST]
    [SS,SE,SC,ST,SI,SO,SN] [SP,
    SO,SI,SN,ST,SE,SR] :-[RVSON,
    LEFT]";S3
56  3050 PRINT"4.  [SL,SA,SS,ST,
    SSPC,SS,SE,SC,ST,SI,SO,SN,SS
    PC,SP,SO,SI,SN,ST,SE,SR] :-[
    RVSON,LEFT]";S4
CF  3060 PRINT"5.  [SN,SE,SX,ST,
    SSPC,SE,SN,ST,SR,SY,SSPC,SP,
    SO,SI,SN,ST,SE,SR,SSPC]:-[RV
    SON,LEFT]";E2
A0  3070 PRINT"6.  [SL,SA,SS,ST]
    [SE,SN,ST,SR,SY] [SP,SO,SI,
    SN,ST,SE,SR] :-[RVSON,LEFT]"
    ;E3
22  3080 PRINT"7.  [SM,SA,SR,SK,
    SE,SR,SSPC,ST,SY,SP,SE,SSPC]
    :-[RVSON,LEFT]";M2
AE  3090 PRINT"[DOWN2,RVSON,SE,S
    N,ST,SE,SR,SSPC,SL,SI,SN,SE,
    SSPC,SN]0.[SSPC,ST,SO,SSPC,S
    B,SE,SSPC,SA,SM,SE,SN,SD,SE,
    SD]"
6A  3100 PRINT"[DOWN,SPC6]8.  [S
    R]ETURN TO [SM]ENU"
10  3110 PRINT"[SPC6]9.  [SS]ECU
    RITY [SA]CCESS"
0F  3120 GETX$:IFX$=""THEN3120
A0  3130 X=VAL(X$)
A7  3140 IFX>7THEN3170
D0  3150 IFX=0THEN3120
D2  3160 IFZ1$="E"ORZ1$="N"ANDX>
    2THEN3530
40  3170 ONXGOTO3250,3280,3330,3
    370,3410,3450,3490,1640,3180
FB  3180 IFZ1$="E"ORZ1$="Y"THEN3
    000
9F  3190 PRINT"[DOWN2,SI]NPUT [S
    P]ASSWORD ";
1F  3200 I3=6:I4=6:GOSUB9200
99  3210 IFI$=P$THENZ1$="Y":GOTO
    3000
C1  3220 PRINT"[DOWN,RVSON,SI,SN
    ,SV,SA,SL,SI,SD] [SP,SA,SS2,
    SW,SO,SR,SD,RVSOFF] - [SS]EC
    URITY [SA]CCESS"
B4  3230 PRINT"WILL NOT BE ALLOW
    ED AT ANY POINT"
AE  3240 Z1$="E":FORX=1TO2000:NE
    XTX:GOTO3000
A4  3250 PRINT"[DOWN,ST]ITLE ";:
    I3=10:I4=1:GOSUB9200
03  3260 T$=I$:RC=R:GOSUB1600:GO
    SUB320:PRINT#1,T$
DE  3270 GOTO3000
38  3280 PRINT"[DOWN,ST]YPE ";:I
    3=1:I4=1:GOSUB9200
13  3290 IFI$="L"ORI$="N"ORI$="T
    "ORI$="[SL]"OR I$="[SN]"ORI$
    ="[ST]"THENPRINT"[UP2]":GOTO
    3310
B7  3300 PRINT"[UP2]":GOTO3280
89  3310 T1$=I$:GOSUB820
10  3320 GOTO3000
43  3330 PRINT"[DOWN,SN]EXT [SS]
    ECTION [SP]OINTER ";S3;" ";
A8  3340 I3=5:I4=1:GOSUB9200
37  3350 S3=I:GOSUB820
27  3360 GOTO3000
67  3370 PRINT"[DOWN,SL]AST [SS]
    ECTION [SP]OINTER ";S4;" ";
E0  3380 I3=5:I4=1:GOSUB9200
54  3390 S4=I:GOSUB820
5F  3400 GOTO3000
E1  3410 PRINT"[DOWN,SN]EXT [SE]
    NTRY [SP]OINTER ";E2;" ";
FB  3420 I3=5:I4=1:GOSUB9200
E0  3430 E2=I:GOSUB820
77  3440 GOTO3000
BF  3450 PRINT"[DOWN,SL]AST [SE]
    NTRY [SP]OINTER ";E3;" ";
30  3460 I3=5:I4=1:GOSUB9200
BD  3470 E3=I:GOSUB820
AF  3480 GOTO3000
5B  3490 PRINT"[DOWN,SM]ARKER [S
    T]YPE ";M2;" ";
CE  3500 I3=3:I4=1:GOSUB9200
58  3510 M2=I:GOSUB820
47  3520 GOTO3000
73  3530 PRINT"[DOWN2,SP]ASSWORD
     ALREADY INPUT WAS INVALID":
    FORX=1TO2000:NEXTX
ED  3540 GOTO3240
C5  3550 R=S1:R1(1)=S1:FORX=2TO1
    1:R1(X)=0:NEXTX:PRINTCHR$(14
    )
D3  3560 PRINT"[CLR,RVSON,SS,SE,
    SC,ST,SI,SO,SN,SSPC,SM,SE,SN
    ,SU]"
0F  3570 IFR=0THEN6530
AB  3580 FORL1=1TO10
16  3590 GOSUB870
D4  3600 T2$(L1)=T$:T3$(L1)=T1$:
    R1(L1+1)=S3
D2  3610 IFLEFT$(T2$(L1),1)="[PI
    ]"THENPRINT"[DOWN,SE,SR2,SO,
    SR] [SZ,SSPC,SN,SO,SSPC,SM,S
    A,SS,ST,SE,SR]":GOTO3650
49  3620 IFR1(L1+1)=1THENR1(11)=
    1:GOTO3650
CA  3630 R=R1(L1+1)
21  3640 NEXTL1
2B  3650 PRINT"[DOWN,SN,SO].";TA
    B(10);"[ST,SI,ST,SL,SE]";TAB
    (30);"[ST,SY,SP,SE]"
C3  3660 PRINT
E2  3670 FORL2=1TO10:IFR1(L2)=0O
    RR1(L2)=1THEN3700
40  3680 PRINTL2;TAB(10);T2$(L2)
    ;TAB(30);T3$(L2)
74  3690 NEXTL2
F5  3700 IFR1(11)=0ORR1(11)=1THE
    NPRINT"[DOWN,RVSON,SC]OMPLET
    E":GOTO3720
6D  3710 PRINT"[DOWN,SM]ORE"
AA  3720 PRINT"[DOWN,SI,SN,SP,SU
    ,ST] [SS,SE,SC,ST,SI,SO,SN]
    ([SZ,SE,SR,SO] [SF,SO,SR] [S
    M,SO,SR,SE])"
68  3730 PRINT"[SPC14](* [SR,SE,
    ST,SU,SR,SN] [ST,SO] [SM,SE,
    SN,SU])"
E4  3740 PRINT"[SPC19]";:I3=2:I4
    =1:GOSUB9200
94  3750 IFI$="*"THEN1640
F4  3760 IFI<>0THEN3790
05  3770 IFR1(11)=0ORR1(11)=1THE
    NPRINT"[UP3]":GOTO3730
1E  3780 R=R1(11):FORX=1TO11:R1(
    X)=0:NEXTX::R1(1)=R:GOTO3560
F6  3790 IFI>L2-1THENPRINT"[UP3]
    ":GOTO3730
BF  3800 R=R1(I)
B2  3810 GOSUB870
78  3820 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN] ";T$;" [SL,SO,SA,
    SD,SE,SD]"
15  3830 FORX=1TO1000:NEXTX
85  3840 GOTO1640
89  3850 E4=0:E5=0:F$="":N1=0:PR
    INTCHR$(14)
33  3860 PRINT"[CLR,RVSON,SC,SR,
    SE,SA,ST,SE,SSPC,SN,SE,SW] [
    SE,SN,ST,SR,SY]"
78  3870 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN]:- ";T$
90  3880 PRINT"[DOWN,SE,SN,ST,SE
    ,SR] [SS,SE,SC,ST,SI,SO,SN]
    [SK,SE,SY] ";
A4  3890 I3=20:I4=2:GOSUB9200
8F  3900 IFE2=0THENE5=R:E4=R:GOT
    O4090
E2  3910 R2=E2
55  3920 GOSUB1130
98  3930 IFK$>=I$THEN4000
1A  3940 IFE4=RTHEN3960
7D  3950 R2=E4:GOTO3920
C2  3960 E4=E:GOSUB1100
84  3970 E4=R:E5=R2
A9  3980 E3=E:GOSUB820
B9  3990 GOTO4090
```

```
8F  4000 IFK$=I$THEN4480
CF  4010 IFE5=RTHEN4060
2C  4020 R8=E5:E5=E:GOSUB1100:R7
    =R2
C2  4030 R2=R8:GOSUB1130:E4=E:GO
    SUB1100
5B  4040 E4=R7:E5=R8
D5  4050 GOTO4090
04  4060 E2=E:GOSUB820
33  4070 E5=E:GOSUB1100
CC  4080 E4=R2:E5=R
39  4090 K$=I$
54  4100 R2=E:N1=0
24  4110 GOSUB430
CA  4120 GOSUB1100:M=M-1:GOSUB14
    90
C2  4130 IFE2<>0THEN4160
3F  4140 E2=R2:E3=R2
61  4150 GOSUB820
FD  4160 IFF$="Y"THENRETURN
77  4170 FORX=1TO10
DF  4180 D$(X)=""
D2  4190 NEXTX:L1=0:D1$=""
93  4200 PRINT"[CLR,RVSON,SE,SN,
    ST,SR,SY,SSPC,SD,SE,ST,SA,SI
    ,SL,SSPC,SI,SN,SP,SU,ST]"
4C  4210 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN]:- [RVSON,LEFT]";T
    $
03  4220 PRINT"[DOWN,SE,SN,ST,SR
    ,SY] [SK,SE,SY]:- [RVSON,LEF
    T]";K$
6B  4230 PRINT"[DOWN,SE,SN,ST,SR
    ,SI,SE,SS] [SU,SS,SE,SD]:- [
    RVSON,LEFT]";L1
6C  4240 PRINT"[DOWN]"
FF  4250 FORX=1TO9
6F  4260 D$(X)=D$(X+1)
08  4270 IFD$(X)=""THEN4290
DD  4280 PRINTL1-(10-X);TAB(4);D
    $(X)
A8  4290 NEXTX
D2  4300 D$(10)=D1$:GOSUB1500
BB  4310 IFL1=0THEN4330
86  4320 PRINTL1;TAB(4);D$(10)
75  4330 PRINT"[DOWN,RVSON]([SE,
    SN,ST,SE,SR,SSPC]'***'[SSPC,
    ST,SO,SSPC,SE,SN,SD,SSPC,SI,
    SN,SP,SU,ST])"
6B  4340 PRINT"[DOWN,SD,SA,ST,SA
    ]"
C8  4350 I3=35:I4=1:GOSUB9200
0D  4360 IFI$="***"ORI$="[CN3]"T
    HENPRINTCHR$(142):GOTO1640
4B  4370 D1$=I$
45  4380 R3=E
0D  4390 GOSUB430
EB  4400 N2=0:GOSUB1300:M=M-1:GO
    SUB1490
41  4410 IFN1<>0THEN4450
63  4420 N1=R3
52  4430 GOSUB1100
66  4440 GOTO4460
80  4450 R8=R3:R3=R9:GOSUB1330:N
    2=R8:GOSUB1300:R3=R8:GOSUB13
    30
5B  4460 L1=L1+1:R9=R3
90  4470 GOTO4200
1E  4480 PRINT"[DOWN2,RVSON,SE,S
    R2,SO,SR,RVSOFF] - [SD,SU,SP
    ,SL,SI,SC,SA,ST,SE] [SK,SE,S
    Y] - [SP,SL,SE,SA,SS,SE]"
BA  4490 PRINT"[SPC8,SR,SE,ST,SR
    ,SY]"
ED  4500 FORX=1TO2000:NEXTX
A6  4510 GOTO3850
9B  4520 PRINTCHR$(14):PRINT"[CL
    R,RVSON,SD,SE,SL,SE,ST,SE] [
    SC,SU,SR2,SE,SN,ST] [SE,SN,S
    T,SR,SY]"
FB  4530 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN] :- [RVSON,LEFT]";
    T$
56  4540 PRINT"[DOWN,SE,SN,ST,SR
    ,SY] [SK,SE,SY] :- [RVSON,LE
    FT]";K$
83  4550 PRINT"[DOWN2,SA,SR,SE]
    [SY,SO,SU] [SS,SU,SR,SE] ([S
    Y]/[SN])"
73  4560 GETX$:IFX$=""THEN4560
A0  4570 IFX$="N"THEN1640
A5  4580 IFX$<>"Y"THEN4560
9C  4590 R$="[PI]"
B6  4600 RC=R2:GOSUB1600:PRINT#1
    ,R$:M=M+1:GOSUB1490:GOSUB154
    0
EE  4610 IFN1=0THEN4690
2A  4620 R3=N1
28  4630 GOSUB1330
5B  4640 RC=R3:GOSUB1600:PRINT#1
    ,R$
A7  4650 GOSUB1560:M=M+1:GOSUB14
    90
15  4660 IFN2=0THEN4690
E7  4670 R3=N2
54  4680 GOTO4630
89  4690 IFF$="Y"THENF$="N":RETU
    RN
2E  4700 R8=E4:R9=E5
30  4710 IFE4<>RTHEN4740
EA  4720 E3=R9:GOSUB820
73  4730 GOTO4750
04  4740 R2=R8:GOSUB1130:E5=R9:G
    OSUB1100
4C  4750 IFR9<>RTHEN4780
10  4760 E2=R8:GOSUB820
4F  4770 GOTO4790
3D  4780 R2=R9:GOSUB1130:E4=R8:G
    OSUB1100
59  4790 E4=0:E5=0:K$="":N1=0:N2
    =0:D$=""
DE  4800 GOTO1640
DB  4810 IFZ1$="E"ORZ1$="Y"THEN4
    830
85  4820 Z1$="N"
71  4830 PRINTCHR$(14):PRINT"[CL
    R,RVSON,SA,SM,SE,SN,SD] [SE,
    SN,ST,SR,SY] - [SH,SE,SA,SD,
    SE,SR]"
D0  4840 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN] :- [RVSON,LEFT]";
    T$
9F  4850 IFK$=""THEN6530
F4  4860 PRINT"[DOWN]1. [SK,SE,S
    Y] :- [RVSON,LEFT]";K$
F4  4870 IFZ1$="N"ORZ1$="E"THEN4
    910
29  4880 PRINT"2. [SN,SE,SX,ST]
    [SE,SN,ST,SR,SY] [SP,SO,SI,S
    N,ST,SE,SR] [RVSON,LEFT]";E4
B0  4890 PRINT"3. [SL,SA,SS,ST]
    [SE,SN,ST,SR,SY] [SP,SO,SI,S
    N,ST,SE,SR] [RVSON,LEFT]";E5
CE  4900 PRINT"4. [SN,SE,SX,ST]
    [SD,SE,ST,SA,SI,SL] [SP,SO,S
    I,SN,ST,SE,SR] [RVSON,LEFT]"
    ;N1
62  4910 PRINT"7. [SA,SM,SE,SN,S
    D] [SD,SE,ST,SA,SI,SL,SS]"
F4  4920 PRINT"8. [SR,SE,ST,SU,S
    R,SN] [ST,SO] [SM,SE,SN,SU]"
C1  4930 PRINT"9. [SS,SE,SC,SU,S
    R,SI,ST,SY] [SA,SC2,SE,SS2]"
07  4940 PRINT"[DOWN,RVSON,SE,SN
    ,ST,SE,SR] [SS,SE,SL,SE,SC,S
    T,SI,SO,SN]"
CE  4950 GETX$:IFX$=""THEN4950
15  4960 X=VAL(X$)
1A  4970 IFX>4THEN5010
F9  4980 IFZ1$="Y"THEN5010
DC  4990 IFX=1THEN5010
84  5000 GOTO4950
10  5010 ONXGOTO5090,5310,5350,5
    390,4950,4950,5430,1640,5020
97  5020 IFZ1$="Y"ORZ1$="E"THEN4
    950
3E  5030 PRINT"[DOWN2,SI,SN,SP,S
    U,ST] [SP,SA,SS2,SW,SO,SR,SD
    ] ";
EA  5040 I3=6:I4=6:GOSUB9200
8E  5050 IFI$=P$THENZ1$="Y":GOTO
    4830
E2  5060 PRINT"[DOWN,RVSON,SI,SN
    ,SV,SA,SL,SI,SD] [SP,SA,SS2,
    SW,SO,SR,SD,RVSOFF] - [SS]EC
    URITY [SA]CCESS [SW]ILL"
41  5070 PRINT"[SN]OT [SB]E [SA]
    LLOWED [SA]T [SA]NY [SP]OINT
    "
12  5080 Z1$="E":FORX=1TO1000:NE
    XTX:GOTO4830
7A  5090 PRINT"[DOWN,SK,SE,SY] "
    ;:I3=20:I4=2:GOSUB9200
20  5100 R8=R2:R0=N1:L1=E5:L2=E4
EE  5110 IFL2=RTHENK1$="[PI]"
CF  5120 R2=L2:GOSUB1130:K1$=K$
7A  5130 IFL1=RTHENK2$="0":GOTO5
    150
11  5140 R2=L1:GOSUB1130:K2$=K$
B8  5150 IFI$=>K2$ANDI$<=K1$THEN
    5290
0D  5160 IFL2<>RTHEN5190
2C  5170 E3=L1:GOSUB820
5C  5180 GOTO5200
2E  5190 R2=L2:GOSUB1130:E5=L1:G
    OSUB1100
7F  5200 IFL1<>RTHEN5230
64  5210 E2=L2:GOSUB820
78  5220 GOTO5240
93  5230 R2=L1:GOSUB1130:E4=L2:G
    OSUB1100
97  5240 R2=R8:RC=R2:GOSUB1600:G
    OSUB320
A5  5250 PRINT#1,"[PI]":M=M+1
AF  5260 GOSUB1540
25  5270 F$="Y":GOSUB3910:F$="N"
    :N1=R0:GOSUB1100
4C  5280 GOTO4830
84  5290 K$=I$:R2=R8:E5=L1:E4=L2
    :N1=R0:GOSUB1100
B0  5300 GOTO4830
39  5310 PRINT"[DOWN,SN,SE,SX,ST
    ] [SE,SN,ST,SR,SY] [SK,SE,SY
    ] [SP,SO,SI,SN,ST,SE,SR] ";E
    4;" ";
F1  5320 I3=5:I4=1:GOSUB9200
A1  5330 E4=I:GOSUB1100
F8  5340 GOTO4830
67  5350 PRINT"[DOWN,SL,SA,SS,ST
    ] [SE,SN,ST,SR,SY] [SK,SE,SY
    ] [SP,SO,SI,SN,ST,SE,SR] ";E
    5;" ";
A9  5360 I3=5:I4=1:GOSUB9200
30  5370 E5=I:GOSUB1100
1F  5380 GOTO4830
49  5390 PRINT"[DOWN,SN,SE,SX,ST
    ] [SE,SN,ST,SR,SY] [SP,SO,SI
    ,SN,ST,SE,SR] ";N1;" ";
BC  5400 I3=5:I4=1:GOSUB9200
6A  5410 N1=I:GOSUB1100
C7  5420 GOTO4830
A7  5430 FORX=1TO11
4E  5440 D$(X)=""
EB  5450 NEXTX:L1=1:R7=R2:R3=N1:
    IFN1=0THENL1=0:GOTO5470
B8  5460 GOSUB1330:D$(1)=D1$
D5  5470 PRINT"[CLR,RVSON,SA,SM,
    SE,SN,SD] [SE,SN,ST,SR,SY] -
     [SD,SE,ST,SA,SI,SL,SS]"
69  5480 PRINT"[DOWN,SS,SE,SC,ST
    ,SI,SO,SN] :- [RVSON,LEFT]";
    T$
34  5490 PRINT"[DOWN,SE,SN,ST,SR
    ,SY] [SK,SE,SY] :- [RVSON,LE
    FT]";K$
2B  5500 PRINT"[DOWN,SC,SU,SR2,S
    E,SN,ST] [SE,SN,ST,SR,SY] [S
```

```
   NJO. ";L1
02 5510 PRINT
14 5520 FORR8=1TO10
89 5530 IFD$(R8)=""THEN5600
77 5540 IFD$(10)=""THEN5560
42 5550 D$(R8)=D$(R8+1):IFR8=10
   THEN5600
DA 5560 IFL1<10THENL2=10:L3=L1-
   R8
C1 5570 IFL1>9THENL3=R8*-1:L2=0
B9 5580 PRINTL1-((10-L2)+L3);TA
   B(5);D$(R8)
83 5590 NEXTR8
CF 5600 IFL1<2THEN5630
81 5610 D$(R8)=D1$
B5 5620 PRINTL1;TAB(5);D1$
73 5630 PRINT"[DOWN,SA]MEND ([S
   Y]/[SN]), 8 - [SM]ENU, 9 [SS
   ]ECURITY [SA]CCSS"
F6 5640 PRINT"[SPC15]6 - [SD]EL
   ETE, 7 - [SI]NSERT"
89 5650 GETX$:IFX$=""THEN5650
7A 5660 IFX$<>"Y"ANDX$<>"N"ANDX
   $<>"8"ANDX$<>"9"ANDX$<>"6"AN
   DX$<>"7"THEN5650
27 5670 IFX$="8"THEN1640
F0 5680 IFX$="9"THEN5820
46 5690 IFX$="N"THEN5780
02 5700 IFX$="6"THEN5870
F8 5710 IFX$="7"THEN5970
91 5720 IFN1=0THEN5650
AB 5730 PRINT"[SE]NTER [SN]EW [
   SD]ETAILS"
E4 5740 I3=35:I4=1:GOSUB9200
CE 5750 D1$=I$:IFL1=1THEND$(R8-
   1)=I$:GOTO5770
EB 5760 D$(R8)=I$
AD 5770 GOSUB1300
6C 5780 IFN2=0THEN5650
23 5790 R7=R3:R3=N2
63 5800 GOSUB1330:IFL1=1THENR7=
   R2
0E 5810 L1=L1+1:GOTO5470
ED 5820 IFZ1$="E"ORZ1$="N"THENS
   650
72 5830 PRINT"[DOWN,SN,SE,SX,ST
   ] [SK,SE,SY] [SP,SO,SI,SN,ST
   ,SE,SR] ";N2;" ";
02 5840 I3=5:I4=1:GOSUB9200:IFI
   $="*"THEN5780
A2 5850 N2=I:GOSUB1300
E7 5860 GOTO5780
D3 5870 IFN1=0THEN5650
57 5880 IFR7<>R2THEN5960
FB 5890 N1=N2:GOSUB1100
9D 5900 RC=R3:GOSUB1600:GOSUB32
   0:PRINT#1,"[PI]"
0D 5910 M=M+1:GOSUB1490
00 5920 GOSUB1560:IFL1=1THEN543
   0
40 5930 D$(R8)="":L1=L1-1
0E 5940 IFN2=0THEN1640
14 5950 R3=N2:GOTO5800
AF 5960 R6=N2:R5=R3:R3=R7:GOSUB
   1330:N2=R6:GOSUB1300:R3=R5:N
   2=R6:GOTO5900
AC 5970 IFN1=0THEN6000
47 5980 R6=N2:N2=E:GOSUB1300:N2
   =R6
7B 5990 GOTO6010
1A 6000 N1=E:GOSUB1100:N2=0
6B 6010 PRINT"[DOWN,SE]NTER [SN
   ]EW [SD]ETAILS":I3=35:I4=1:G
   OSUB9200:D1$=I$
FE 6020 R3=E:GOSUB430:GOSUB1300
   :M=M-1:GOSUB1490:L1=L1+1
88 6030 IFL1=1THEND$(1)=D1$
96 6040 GOTO5470
2B 6050 R6=N2:N2=E:GOSUB1300:N2
   =R6:GOTO6010
04 6060 FORX=1TO10:D1$(X)="":NE
   XTX:L2=0:R3=N1:X2$="[SM]ORE"
   :L3=0
6F 6070 FORL1=1TO10
A7 6080 IFN1=0THEN6140
D1 6090 GOSUB1330
F0 6100 D1$(L1)=D1$:L2=L2+1
98 6110 IFN2=0THENX2$="[SC]OMPL
   ETE":GOTO6140
92 6120 R3=N2
E6 6130 NEXTL1
32 6140 PRINTCHR$(14):PRINT"[CL
   R,RVSON,SV,SI,SE,SW] [SE,SN,
   ST,SR,SY]"
C1 6150 IFN1=0THEN6530
79 6160 PRINT"[DOWN,SS,SE,SC,ST
   ,SI,SO,SN] :- [RVSON,LEFT]";
   T$;"[RVSOFF] [ST,SY,SP,SE] :
   - [RVSON,LEFT]";T1$
2F 6170 PRINT"[DOWN,SK,SE,SY] :
   - [RVSON,LEFT]";K$
C7 6180 PRINT"[DOWN,SC]URRENT [
   SR]ECORD [SN]O. [RVSON]";L2-
   L3;"[DOWN]":X1=0:X2=L2:IFL2>
   =10THENX1=10:X2=0
9B 6190 FORX=1TO10
81 6200 IFD1$(X)=""THEN6230
A7 6210 PRINTL2-(X1-X+X2);TAB(4
   );D1$(X)
3E 6220 NEXTX
7F 6230 PRINT"[DOWN]";X2$
EC 6240 IFN2<>0THEN6280
4F 6250 PRINT"[DOWN,SP]RESS ANY
    KEY TO RETURN TO MENU"
03 6260 GETX$:IFX$=""THEN6260
AA 6270 GOTO1640
5B 6280 PRINT"[DOWN,SM,SO,SR,SE
   ]? ([SY]/[SN])"
EC 6290 GETX$:IFX$=""THEN6290
C1 6300 IFX$="N"THEN1640
17 6310 IFX$="Y"THEN6330
4E 6320 GOTO6290
C0 6330 FORL1=1TO5
A7 6340 D1$(L1)=D1$(L1+5)
81 6350 NEXTL1
7B 6360 FORL1=6TO10:D1$(L1)="":
   NEXTL1
A3 6370 FORL1=6TO10
68 6380 L2=L2+1:R3=N2:GOSUB1330
D0 6390 D1$(L1)=D1$
B9 6400 IFN2=0THENX2$="[SC]OMPL
   ETE":L2=L2+10-L1:L3=10-L1:GO
   TO6140
DC 6410 NEXTL1
2F 6420 GOTO6140
9C 6430 PRINTCHR$(14):PRINT"[CL
   R,RVSON,SE,SN,ST,SR,SY] [SM,
   SE,SN,SU]"
DB 6440 R2=E2:R1(1)=E2:FORX=2TO
   11:R1(X)=0:T3$(X-1)="":NEXTX
53 6450 PRINT"[DOWN,SD]O YOU RE
   QUIRE PATTERN SEARCH ([SY]/[
   SN])"
C2 6460 GETX$:IFX$<>"Y"ANDX$<>"
   N"THEN6460
50 6470 IFX$="N"THENS$="":GOTO6
   520
41 6480 PRINT"[DOWN,SI]NPUT SEA
   RCH KEY (USE '!' FOR WILD"
6F 6490 PRINT"CHARCTERS."
20 6500 I3=20:I4=2:GOSUB9200
B7 6510 S$=I$
EF 6520 IFE2<>0THEN6560
30 6530 PRINT"[DOWN,SN]O [SD]ET
   AIL [SR]ECORDS [SO]N [SF]ILE
   "
E9 6540 FORX=1TO1500:NEXTX
A5 6550 GOTO1640
AF 6560 PRINT"[CLR,RVSON,SE,SN,
   ST,SR,SY] [SM,SE,SN,SU]":PRI
   NT"[DOWN,SS,SE,SC,ST,SI,SO,S
   N] :- [RVSON,LEFT]";T$;"[RVS
   OFF] [ST,SY,SP,SE] :- [RVSON
   ,LEFT]";T1$
CD 6570 PRINT"[DOWN,SN]O.";TAB(
   10);"[SK]EY":PRINT
6D 6580 FORR9=1TO10
DD 6590 GOSUB1130
BA 6600 IFS$=""THEN6660
DC 6610 GOSUB240
96 6620 IFF1$="Y"THEN6650
53 6630 R1(R9+1)=E4:IFE4=RTHENR
   1(11)=1:GOTO6710
D5 6640 R2=E4:GOTO6590
6C 6650 R1(R9)=R2
C0 6660 T3$(R9)=K$
93 6670 R1(R9+1)=E4
B0 6680 IFR1(R9+1)=RTHENR1(11)=
   1:GOTO6710
DA 6690 R2=R1(R9+1)
2F 6700 NEXTR9
C1 6710 FORR8=1TO10
BC 6720 IFR1(R8)=0THEN6770
30 6730 IFR1(R8)=RTHEN6770
A0 6740 IFT3$(R8)=""THEN6770
DA 6750 PRINTR8;TAB(10);T3$(R8)
6A 6760 NEXTR8
47 6770 IFR1(11)=0ORR1(11)=RORR
   1(R9+1)=RORR1(11)=1THENPRINT
   "[DOWN,RVSON,SC]OMPLETE":GOT
   O6790
D3 6780 PRINT"[DOWN,SM]ORE"
71 6790 PRINT"[DOWN,SI,SN,SP,SU
   ,ST] [SS,SE,SL,SE,SC,ST,SI,S
   O,SN] ([SZ,SE,SR,SO] FOR [SM
   ]ORE)"
92 6800 PRINT"[SPC16](* FOR [SM
   ]ENU)"
BA 6810 PRINT"[SPC20]";:I3=2:I4
   =1:GOSUB9200
A2 6820 IFI$="*"THEN1640
E1 6830 IFI<>0THEN6880
40 6840 IFR1(11)=0ORR1(11)=1THE
   N6890
74 6850 R2=R1(11)
C3 6860 FORX=1TO11:R1(X)=0:NEXT
   X:R1(1)=R2
E0 6870 GOTO6560
AC 6880 IFI<=R8-1THEN6910
50 6890 PRINT"[UP3]"
85 6900 GOTO6800
99 6910 R2=R1(I)
11 6920 GOSUB1130
87 6930 N2=0:D$=""
CC 6940 FORX=1TO10:D1$(X)="":NE
   XTX
75 6950 PRINT"[DOWN,SE,SN,ST,SR
   ,SY] ";K$;" [SL,SO,SA,SD,SE,
   SD]"
F7 6960 FORX=1TO1500:NEXTX
5F 6970 GOTO1640
8B 6980 PRINT"[CLR]":PRINTCHR$(
   14):PRINT"[RVSON,SL,SA,SB,SE
   ,SL,SS]"
F7 6990 IFT1$="L"ORT1$="[SL]"TH
   EN7030
4A 7000 PRINT"[DOWN,SM,SA,SR,SK
   ,SE,SR] [ST,SY,SP,SE] [SN,SO
   ,ST] [SL,SA,SB,SE,SL]"
44 7010 FORX=1TO1000:NEXTX
0D 7020 GOTO1640
6D 7030 ONX1GOTO7040,7360
65 7040 PRINT"[CLR,SA,SL2] [SL,
   SA,SB,SE,SL] [SP,SR,SI,SN,ST
   ]"
21 7050 PRINT"[DOWN,SS,SE,SC,ST
   ,SI,SO,SN] :- [RVSON,LEFT]";
   T$
CC 7060 PRINT"[DOWN]":IFF$="Y"T
   HENRETURN
9B 7070 GOSUB8670:R7=18
56 7080 IFE2=0THEN7340
9A 7090 R2=E2
DD 7100 GOSUB1130
A4 7110 IFN1<>0THEN7140
56 7120 PRINT"[DOWN,SN,SO] [SL,
   SA,SB,SE,SL,SS] [SF,SO,SR] [
   SE,SN,ST,SR,SY] ";K$;"[UP2]"
```

```
   :FORZ1=1TO1000:NEXTZ1
87 7130 GOTO7260
91 7140 R3=N1
AF 7150 FORS=1TOL%
C1 7160 GOSUB1330:D$(10)=D1$:GO
   SUB1500:L$(S)=D$(10)
DE 7170 IFN2=0THEN7210
5F 7180 R3=N2
03 7190 NEXTS
92 7200 S=L%
05 7210 IFN2=0ORS<L%THEN7250
5A 7220 IFR7=18THENF$="Y":GOSUB
   7040:F$="N":R7=0
A3 7230 R7=R7+1
C4 7240 PRINT"[DOWN,SW,SA,SR,SN
   ,SI,SN,SG] [SL,SA,SB,SE,SL]
   [SO]/[SF,SL,SO,SW] ";K$
69 7250 GOSUB8860
E0 7260 IFF1$="Y"THENF1$="N":GO
   TO7300
A5 7270 IFE4=RTHEN7300
D3 7280 R2=E4
A7 7290 GOTO7100
32 7300 PRINT"[DOWN3,SL,SA,SB,S
   E,SL] [SP,SR,SI,SN,ST] [SC,S
   O,SM,SP,SL,SE,ST,SE]"
DB 7310 FORX=1TO1000:NEXTX
A6 7320 GOSUB1480
A2 7330 GOTO1640
3E 7340 PRINT"[DOWN,SN,SO] [SL,
   SA,SB,SE,SL,SS] [ST,SO] [SP,
   SR,SI,SN,ST]"
E2 7350 GOTO7310
A5 7360 PRINT"[CLR,SS,SI,SN,SG,
   SL,SE] [SL,SA,SB,SE,SL] [SP,
   SR,SI,SN,ST]"
34 7370 PRINT"[DOWN,SS,SE,SC,ST
   ,SI,SO,SN] :- [RVSON,LEFT]";
   I$
F1 7380 PRINT"[DOWN,SK,SE,SY] :
   - [RVSON,LEFT]";K$
B0 7390 F1$="Y":GOSUB8670
16 7400 GOTO7110
27 7410 PRINT"[CLR]":PRINTCHR$(
   14):IFE2=0THENPRINT"[CLR,RVS
   ON,SA,SL2] [SR,SE,SP,SO,SR,S
   T]":GOTO6530
19 7420 GOSUB1470
73 7430 PRINT#4:PRINT#4," [SS,S
   E,SC,ST,SI,SO,SN] :-  [LEFT]
   ";T$;"  [ST,SY,SP,SE]  [LEFT
   ]";T1$
0B 7440 CLOSE4:GOSUB1460
0B 7450 PRINT"[CLR,RVSON,SA,SL2
   ] [SR,SE,SP,SO,SR,ST]":PRINT
   "[DOWN,SS,SE,SC,ST,SI,SO,SN]
    :- [RVSON,LEFT]";T$;"[RVSOF
   F] [ST,SY,SP,SE] [RVSON]";T1
   $
D9 7460 IFT1$="T"ORT1$="[ST]"TH
   EN7500
06 7470 PRINT"[DOWN] 1 [SO,SR]
   2 [SU,SP]"
BB 7480 GETX$:IFX$=""THEN7480
61 7490 R8=VAL(X$):IFR8=0ORR8>2
   THEN7480
12 7500 IFE2=0THEN8480
14 7510 R2=E2
73 7520 GOSUB1130
EC 7530 L$(1)=K$
A3 7540 R3=N1
CE 7550 IFT1$="T"ORT1$="[ST]"TH
   EN8290
8D 7560 IFR8=1THEN8070
47 7570 IFN1=0THENL$(2)="[SN,SO
   ] [SE,SN,ST,SR,SI,SE,SS]":S=
   2:GOTO7640
34 7580 FORS=2TO50
71 7590 GOSUB1330:D$(10)=D1$:GO
   SUB1500:L$(S)=D$(10)
8A 7600 IFN2=0THEN7640
6A 7610 R3=N2
B6 7620 NEXTS
F9 7630 PRINT"[DOWN,SW,SA,SR,SN
   ,SI,SN,SG] - [SS]TORE [SO]VE
   RFLOW [RVSON]";K$
0A 7640 IFE4=RTHENF2$="Y":GOTO7
   670
98 7650 R2=E4:F2$="N"
87 7660 GOSUB1130
BF 7670 PRINT#4
16 7680 OPEN7,4,2
0F 7690 X$="AAAAA AAAAAAAAAAAA
   AAAAAAAAAAAAAAAAAA"
9B 7700 X$=X$+"[SPC6]AAAAA AAAA
   AAAAAAAAAAAAAAAAAAAAAAAAAAAA
   "
3C 7710 PRINT#7,X$:CLOSE7
30 7720 IFF2$="Y"THEN7750
3C 7730 PRINT#4,"[SK,SE,SY]:- "
   ,CHR$(29),L$(1),CHR$(29),"[S
   K,SE,SY]:- ",CHR$(29),K$
43 7740 GOTO7760
53 7750 PRINT#4,"[SK,SE,SY]:- "
   ,CHR$(29),L$(1)
F4 7760 R9=1:X$=""
E7 7770 R3=N1:N2=N1
66 7780 IFN2=0THENX$="[SN,SO] [
   SE,SN,ST,SR,SI,SE,SS]":GOTO7
   810
5E 7790 IFN2=0THEN7810
72 7800 GOSUB1330:D$(10)=D1$:GO
   SUB1500
4D 7810 R9=R9+1
8A 7820 OPEN7,4,2
5F 7830 R$="99 AAAAAAAAAAAAAAA
   AAAAAAAAAAAAAAAAAA"
39 7840 R$=R$+" 99 AAAAAAAAAAA
   AAAAAAAAAAAAAAAAAAAAAAA"
EA 7850 PRINT#7,R$:CLOSE7
BF 7860 IFF2$="Y"ANDR9>STHEN826
   0
EF 7870 IFF2$="Y"THEN7940
49 7880 IFR9>SANDN2=0ANDF3$="Y"
   THENF3$="N":GOTO8260
7E 7890 IFN2<>0ANDR9<=STHEN7980
07 7900 IFR9>STHEN7960
50 7910 IFX$=""THEN7940
23 7920 PRINT#4,R9-1,L$(R9),CHR
   $(29),R9-1,;D1$:X$="":F3$="Y
   "
11 7930 GOTO7810
EE 7940 PRINT#4,R9-1,L$(R9)
24 7950 GOTO7810
53 7960 PRINT#4,R9-1,CHR$(160)C
   HR$(29),R9-1,D$(10):IFN2=0TH
   ENF3$="Y"
CF 7970 GOTO7990
38 7980 PRINT#4,R9-1,L$(R9),CHR
   $(29),R9-1,D$(10)
E4 7990 R3=N2
43 8000 GOTO7790
9F 8010 PRINT"[DOWN5,RIGHT12,SP
   ,SR,SI,SN,ST] [SC,SO,SM,SP,S
   L,SE,ST,SE,SD]"
07 8020 CLOSE4
FD 8030 PRINTCHR$(14):OPEN5,4,8
   :PRINT#5:CLOSE5
AF 8040 OPEN6,4,10:PRINT#6:CLOS
   E6
45 8050 FORX=1TO1500:NEXTX
99 8060 GOTO1640
71 8070 PRINT#4
85 8080 OPEN7,4,2
FA 8090 X$="AAAAA AAAAAAAAAAAA
   AAAAAAAAAAAAAAAAAA"
50 8100 X$=X$+"[SPC6]AAAAA AAAA
   AAAAAAAAAAAAAAAAAAAAAAAAAAAA
   "
ED 8110 PRINT#7,X$:CLOSE7
81 8120 PRINT#4,"[SK,SE,SY]:-",
   CHR$(29),K$
B3 8130 PRINT#4:R9=1
C9 8140 OPEN7,4,2
7C 8150 R$="99 AAAAAAAAAAAAAAA
   AAAAAAAAAAAAAAAAAA"
5E 8160 R$=R$+" 99 AAAAAAAAAAA
   AAAAAAAAAAAAAAAAAAAAAAA"
23 8170 PRINT#7,R$:CLOSE7
D1 8180 IFR3=0THENPRINT#4,1,"[S
   N,SO] [SE,SN,ST,SR,SI,SE,SS]
   ":GOTO8260
11 8190 GOSUB1330:D$(10)=D1$:GO
   SUB1500
08 8200 PRINT#4,R9,D$(10):R9=R9
   +1
7E 8210 IFN2=0THEN8260
76 8220 R3=N2:GOTO8190
1A 8230 IFN2=0THEN8260
E7 8240 R3=N2
59 8250 GOTO8200
0E 8260 IFE4=RTHEN8010
15 8270 R2=E4
7F 8280 GOTO7520
B2 8290 OPEN7,4,2
73 8300 X$="AAAAA AAAAAAAAAAAA
   AAAAAAAAAAAAAAAAAA"
F6 8310 PRINT#7,X$:CLOSE7
6C 8320 PRINT#4
B8 8330 PRINT#4,"[SK,SE,SY]:- "
   ,CHR$(29),K$
60 8340 PRINT#4
B6 8350 OPEN7,4,2
B3 8360 R$="AAAAAAAAAAAAAAAAAA
   AAAAAAAAAAAAAAAA"
CB 8370 R$=R$+"AAAAAAAAAAAAAAAA
   AAAAAAAAAAAAAAAAAA":PRINT#7
   ,R$:CLOSE7
CD 8380 IFR3=0THENPRINT#4,"[SN,
   SO] [SE,SN,ST,SR,SI,SE,SS]":
   GOTO8260
D2 8390 GOSUB1330
21 8400 D$(10)=D1$
26 8410 IFN2=0THEN8440
FD 8420 R3=N2:GOSUB1330
BD 8430 D$(10)=D$(10)+D1$
5D 8440 GOSUB1500:PRINT#4,D$(10
   )
61 8450 IFN2=0THEN8260
7C 8460 R3=N2
CD 8470 GOTO8380
C0 8480 PRINT"[DOWN,RIGHT10,SN,
   SO] [SE,SN,ST,SE,SR,SI,SE,SS
   ] [SP,SR,SE,SS,SE,SN,ST]"
CC 8490 GOTO8020
31 8500 PRINTCHR$(14):PRINT"[CL
   R,RVSON,SS,SI,SN,SG,SL,SE] [
   SR,SE,SP,SO,SR,ST]"
5E 8510 GOSUB1470
EE 8520 PRINT"[DOWN,SS,SE,SC,ST
   ,SI,SO,SN]:- [RVSON,LEFT]";T
   $;"[RVSOFF] [ST,SY,SP,SE] :-
    [RVSON,LEFT]";T1$
1F 8530 PRINT#4
41 8540 PRINT#4,"[SK,SE,SY]:- [
   RVSON]";K$:PRINT#4
5D 8550 IFN1=0THEN8010
A3 8560 R3=N1
97 8570 GOSUB1330
00 8580 R$=D1$
D9 8590 IFT1$<>"T"ANDT1$<>"[ST]
   "THEN8630
FA 8600 IFN2=0THEN8630
20 8610 R3=N2:GOSUB1330
22 8620 R$=R$+D1$
25 8630 PRINT#4,R$
F2 8640 IFN2=0THENCLOSE5:GOTO80
   10
9E 8650 R3=N2
88 8660 GOTO8570
B6 8670 OPEN4,4,0:OPEN5,4,7:PRI
   NT#5
DE 8680 CLOSE5
C3 8690 IFR=F8THENRETURN
88 8700 INPUT"[DOWN,SN]O. OF LI
   NES PER LABEL ";L%:F8=R:F9$=
   "N"
FA 8710 INPUT"[DOWN,SN]O. OF CH
   ARACTERS PER LINE ";C%
8F 8720 INPUT"[DOWN,SN]O. OF LI
   NES BETWEEN LABELS ";S%
45 8730 PRINT"[DOWN,SI]S LINE C
   ENTERING REQUIRED ([SY]/[SN]
   )"
```

```
70 8740 GETL$:IFL$<>"N"ANDL$<>"Y"THEN8740
20 8750 PRINT"[DOWN,SI]S CHARACTER CENTERING REQUIRED ([SY]/[SN])"
65 8760 GETC$:IFC$<>"N"ANDC$<>"Y"THEN8760
E2 8770 PRINT"[DOWN,SA]RE THESE ADDRESS LABELS ([SY]/[SN])":FORX=1TO9:S$(X)="":NEXTX
EE 8780 GETA$:IFA$="N"THEN8830
DF 8790 IFA$<>"Y"THEN8780
0F 8800 S$(1)="":S$(2)="  ":S$(3)="[SPC4]":S$(4)="[SPC6]":S$(5)="[SPC8]"
95 8810 S$(6)="[SPC10]":S$(7)="[SPC12]":S$(8)="[SPC14]"
C9 8820 S$(9)="[SPC16]"
0E 8830 S=1:D$="":FORX=1TOC%:D$=D$+"[SX]":NEXTX
D6 8840 R8=INT(C%/35):R9=C%-(R8*35):R$=""
3A 8850 RETURN
FC 8860 IFF9$="Y"THEN8950
B8 8870 F9$="Y":INPUT"[DOWN,SH]OW MANY LABELS ";N%
47 8880 PRINT"[DOWN,SI]S LINE UP REQUIRED ([SY]/[SN])[UP2]"
5F 8890 GETX$:IFX$="N"THENPRINT"[DOWN2]":GOTO8950
50 8900 IFX$<>"Y"THEN8890
7F 8910 FORX3=1TOL%:PRINT#4,D$:NEXTX3
CF 8920 IFS%=0THEN8940
BE 8930 FORX3=1TOS%:PRINT#4:NEXTX3
8F 8940 GOTO8880
F7 8950 X5=0
28 8960 X5=X5+1
60 8970 IFX5>N%THEN9180
3B 8980 IFL%=STHEN9010
55 8990 IFL$="N"THEN9010
70 9000 X6=INT((L%-S)/2):FORX7=1TOX6:PRINT#4:NEXTX7
D7 9010 FORX3=1TOS
2D 9020 IFX5>1THEN9110
52 9030 IFA$="Y"THENR9=INT((C%-20)/2):R$="":GOTO9070
8A 9040 IFC$="N"THEN9090
CB 9050 R8=LEN(L$(X3))
BF 9060 R9=INT((C%-R8)/2):R$=""
30 9070 FORX2=1TOR9:R$=R$+" ":NEXTX2
D1 9080 L$(X3)=R$+L$(X3)
8F 9090 IFX3>9THENL$(X3)=S$(9)+L$(X3):GOTO9110
92 9100 L$(X3)=S$(X3)+L$(X3)
AD 9110 PRINT#4,L$(X3)
AA 9120 NEXTX3
F4 9130 IFS=L%THEN9150
FB 9140 FORX=1TOL%-S-X6:PRINT#4:NEXTX
AE 9150 IFS%=0THEN9170
BF 9160 FORX3=1TOS%:PRINT#4:NEXTX3
F3 9170 GOTO8960
C6 9180 FORX3=1TO50:L$(X3)="":NEXTX3
87 9190 RETURN
5E 9200 REM ********************
DB 9210 REM *SUBROUTINE TO INPUT*
DD 9220 REM *A ALPHANUMERIC DATA*
1D 9230 REM *FIELD.            *
99 9240 REM ********************
81 9250 I$=""
55 9260 PRINT"[CM]";
9F 9270 FORI=1TOI3
E6 9280 PRINT"[C@]";
35 9290 NEXTI
F3 9300 PRINT"[CG]";
A0 9310 FORI=1TOI3+1
F7 9320 PRINT"[LEFT]";
5D 9330 NEXTI
93 9340 GETI1$:IFI1$=""THEN9340
EB 9341 IFI1$=","THEN9340
2C 9342 IFI1$=CHR$(44)THEN9340
17 9350 IFI1$=CHR$(13)GOTO9550
D8 9360 IFI1$="]"THEN9510
35 9370 IFI1$=CHR$(20)THEN9440
79 9380 IFI1$<>"["THEN9490
0A 9390 I=LEN(I$)
CE 9400 FORI1=1TOI+1
65 9410 PRINT"[LEFT]";
8A 9420 NEXTI1
E3 9430 GOTO9250
95 9440 IFI$=""THEN9340
92 9450 IFLEN(I$)=1THENI$="":GOTO9470
D4 9460 I$=LEFT$(I$,LEN(I$)-1)
FC 9470 I1$="[LEFT,C@,LEFT]"
2B 9480 GOTO9530
76 9490 IFI1$<" "THEN9340
D9 9500 IFI1$>"Z"ANDI1$<"[CJ]"GOTO9340
38 9510 IFLEN(I$)=I3GOTO9340
B2 9520 I$=I$+I1$:IFI1$="]"THENI1$="."
59 9530 PRINTI1$;
6E 9540 GOTO9340
06 9550 IFLEN(I$)<I4THEN9340
E4 9560 PRINT
C5 9570 I=VAL(LEFT$(I$,1))
34 9580 IFI=0THENRETURN
29 9590 I=VAL(I$):RETURN
```

EXTENDING BASIC *continued from p.30*

routine, because it isn't a subroutine ending with an RTS, as we will see in a minute.

First the routine tests to see if there's enough space on the stack to do a GOSUB. The routine needs three bytes space on the stack. This value is loaded into the accumulator (line 1180), and then the test itself is made (line 1190). If thee isn't enough space left on the stack, routine $A3FB ends with the dreaded error report "OUT OF MEMORY".

Otherwise, five bytes are pushed onto the stack: the first two bytes point at the parameter after the GOSUB command. Remember, $7A/7B are part of the CHARGET routine and, as such, point to the next character. This is saved by pushing it onto the stack (lines 1200-1230).

Next, the current line number which is contained in zero page $39/3A is pushed onto the stack (lines 1240-1270). Finally, the token code for GOSUB ($8D) is pushed onto the stack (lines 1280-1290).

Whenever a RETURN command is given, the computer searches for a block of data like this, starting with $8D. Then the stack pointer is modified so that it points to this block, and the block is pulled back from the stack. Now the Basic program flow can jump back to the proper line, and the character pointer in $7A/7B will again point at the parameter in that line. All that has to be done now is find the beginning of the next line so that program execution can continue as from there.

In line 1300 of our routine we call the GOTO routine, which makes the Basic program jump to the subroutine which has been called. Finally (line 1310), we jump to the ROM routine at $A7AE which executes the next command. Please note that we jump to this routine as opposed to not returning to it, as with the other routines. Can you guess why?

In lines 1200-1290 of our GOSUB routine we have pushed five bytes onto the stack. This changes the stack pointer by five locations! Therefore it would be impossible to use an RTS to return us to the proper routine in line 400.

In the next article, I'd like to expand on GOTO and GOSUB by developing a labeling system for Basic which works in the same way most assembler programs. That is, you can declare a label and then use GOTO and GOSUB to jump to the line contained in that label.

Figure 1: ROM Routines Used

$AD8A	Evaluate an expression and put the result into flp accu #1.
$B7F7	Convert value in flp accu #1 into integer and put result into zero page $14/15 and Y-register/accumulator.
$A8A3	Execute GOTO line number contained in $14/15.
$A3FB	Test to see if there's enough room on the stack to push the number of bytes onto the stack which is contained in the accumulator. If not, give error report "OUT OF MEMORY".

YOUR COMMODORE

Lineage: 58p per word. (including VAT)
Semi display: £10.95 plus VAT per single column centimetre minimum 2cm. Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid. Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01-437 0699

Send your requirements to
ANNA EDDIE
CLASSIFIED DEPARTMENT
ASP LTD 1 GOLDEN SQUARE
LONDON W1R 3AB.

SOFTWARE

YOUR COMMODORE SPECIALS

TECH DRAW 64 - A comprehensive technical illustration aid for C64 owners for both disk and tape. Commands available include LINE DRAW, RAYS, BOX, CIRCLE, ARC, ELLIPSE, FREEHAND, FILL, HATCH & TEXT. Wipe copy, move and rotate, save and load or print your drawing all for £8.95 tape or disk. Order Codes TD64T (tape) TD64D (disk)

Full details in Your Commodore July 1987.

SPEEDY ASSEMBLER

Your Commodore's very own Assembler, a 100% memory resident program for loading from tape or disk, featured in the Your Commodore Machine Code Series and now the Y.C. standard assembler only £9.95 Tape or Disk. Order Codes YSPASSD (disk) YSPASSR (tape)

Order from Your Commodore Readers Services at 9, Hall Road, Maylands Wood Estate, Hemel, Hempstead Herts HP2 7BH. Please make cheques payable to A.S.P. Ltd., or telephone your Access/Visa order on 0442 41221

PUBLIC DOMAIN SOFTWARE FOR THE C64/128.

Just in from the USA 150 disks full of games, business programmes, utilities, educational, music, graphics + more! *Send SAE or phone for our free catalogue.*

Kingsway Computer Services
Dept 4, 140 Rushdale Road, Sheffield S8 9QE. Tel: 0742 588429

ONLY POOLS AND HORSES

(and fixed odds)

No gimmicks. Programmes for punters from the *GENUINE* experts first grade software at sensible prices. If you want to
WIN REGULARLY
phone 051 - 336 - 2668 or send SAE to Boxoft, Y.C. 65 Allan's Meadow, Neston, S. Wirral, L64 9SQ. Commodore 64/128

FOR SALE

TELECODE Finder. Locate STD codes rapidly. Intercode Finder. Locate international codes easily. Psychology plus. Analyse your character traits. usable with/without Geos, fully documented. £8.00 C64/C128 disk only.

WHY NOT FAX US DETAILS OF THE ADVERT YOU REQUIRE OUR NUMBER IS 01-437 1006

SPECIAL OFFER

T.C. COMPUTERS

Tel: Wantage 02357 60177

AMIGA A500 + T.V. Modulator . . . £349.00
AMIGA A500 + 1084S (New Stereo) . . . £599.00
AMIGA A500 + 1Mg Disk Drive . . . £439.00
AMIGA A500 SUPER PACK . . . £399.00
AMIGA System 500 Colour . . . £729.00

Also Atari ST and PC Compatibles including Amstrad available.
ENQUIRIES WELCOME

Please send S.A.E. for details:-
NOW AT:-
12A Barnards Way, Charlton Heights, Wantage, Oxon OX12 7EB

BM64/128 Software Library Latest titles, 7 day hire, all tapes £1.60 disks £2. For list SAE to Mr C. Anderson, 14 Chippendale Court, Belfast BT10 0DU.

YOUR COMMODORE HAS RATES TO SUIT ALL ADVERTISERS LARGE OR SMALL

SERVICES

COMMODORE SUPPLIES

C64 Power Pack . . . ££17.95
64 Data Rec . . . £19.95
C3N (C.16) Data Rec . . . £19.95
C16 Power pack . . . £12.95
Neos Mouse & Cheese . . . £19.95

Prices include VAT & P&P. Chq to:
OMINDALE LTD
23 Curzon Street, Derby.
Tel: 0332 291219

MAIL ORDER

JOHNS COMPUTER STORE

(Home & Business Systems)

COMMODORE AMIGA A500 (incl. Modulator & Software pack) . . . £369.00
COMMODORE C64 HOLLYWOOD PACK . . . £149.00
COMMODORE ENTERTAINMENT PACK . . . £189.00

Phone for details

For further mail order offers (All prices inclusive of VAT) contact:-

J.C.S. 93 Graham Street, Airdrie, Scotland ML6 6BU
Tel: 0236 69874

DISCUSS YOUR ADVERTISING WITH ANNA EDDIE 01 - 437 0626

DISKS

Why Take Risks?...Buy **Branded** disks.

GOLDSTAR 3½" DS/DD DISKS
Boxed in 10's, 100% error free, Branded, Fully Guaranteed, Includes Labels.

£8.30 + VAT Per box of 10

PLEASE ADD 87p + VAT FOR P&P
Goods sent same day as order received.
P.V. TUBES (MAIL ORDER DEPT.)
104 Abbey St., Accrington, Lancs, BB5 1EE
Tel: (0254) 36521/32611/390936
Fax: 0254-395361
PARTNERS: Mrs. S. Cucknell/Mr. B. Cucknell

Repairs Guide

CALL
ANNA EDDIE
ON
01 — 437 0699

PROBLEMS WITH YOUR COMMODORE ?

FOR FAST, RELIABLE AND PROFESSIONAL REPAIRS AT COMPETITIVE PRICES

P M ENGINEERING

UNIT 260, MILTON SCIENCE PARK, CAMBRIDGE CB4 4WE

ST. IVES (0480) 61394

We can also supply Hardware, Software, Blank Disks, Accessories and Spare parts.
C64 Repairs £19.95 inc. VAT (p&p FREE)

BREAK

Readers Problems

Though the Commodore 64 is one of the world's most popular microcomputers, it can be very difficult to find speecific information about your particular machine.

At the *Your Commodore* office we receive literally hundreds of letters from you, our readers, on a wide range of subjects ranging from the simple 'Can you give me the telephone number for ...', to the more complex 'I'm trying to write a program that uses a split screen. How do I do it?'

Unfortunately, the volume of mail received has become so great that it is impossible to answer every letter and still mange to publish a magazine each month.

For this reason we have felt it necessary to produce a number of guidelines for getting information from us.

1) We cannot guarantee to answer every letter sent to the magazine. Should it become apparent that a number of readers are suffering from the same problem, then we will reply to the letter via the Letters page.

2) A new helpline has been set up. This will be open for your queries on Tuesday and Thursday afternoons between 2.00pm and 4.00pm. We will not be able to deal with your telephone queries at any other time. If our technical adviser is not available when you ring, then a message will be taken

3) If you are having problems with one of our listings, can you please let us know in writing. This will enable us to see if a number of people are having the same problem. When a common problem becomes apparent with a program, then a correction sheet will be issued. Enclose a self-addressed, stamped envelope and we will send you a copy of the correction sheet as soon as it is available.

We are sorry that it has become necessary to instigate these rules. However, we are sure that you will agree with us that the more time that we can spend making *Your Commodore* the most informative magazine around, the better.

For program queries write to:
Program Corrections
Your Commodore
1 Golden Square
London
W1R 3AB
If you wish to telephone then call:
01-437 0626 Extn 212

Commodore Where Are You?

At the *Your Commodore* office we are repeatedly asked for the address and telephone number of Commodore U.K. Many people, after referring to their computer manuals, believe them to be based in Corby.

The Commodore plant at Corby was closed down some time ago. Reproduced here you will find the correct address for Commodore U.K.

We suggest that you write this correct address in the front of your computers manual for future reference.

Commodore Business Machine, (UK),
Commodore House,
The Switchback,
Gardner Road,
Maidenhead,
Berks SL6 7XA.

Tel: (0628) 770088

William Tell

If you have been typing in the William Tell program over the last few months you will be pleased to note that it is completed in this issue .

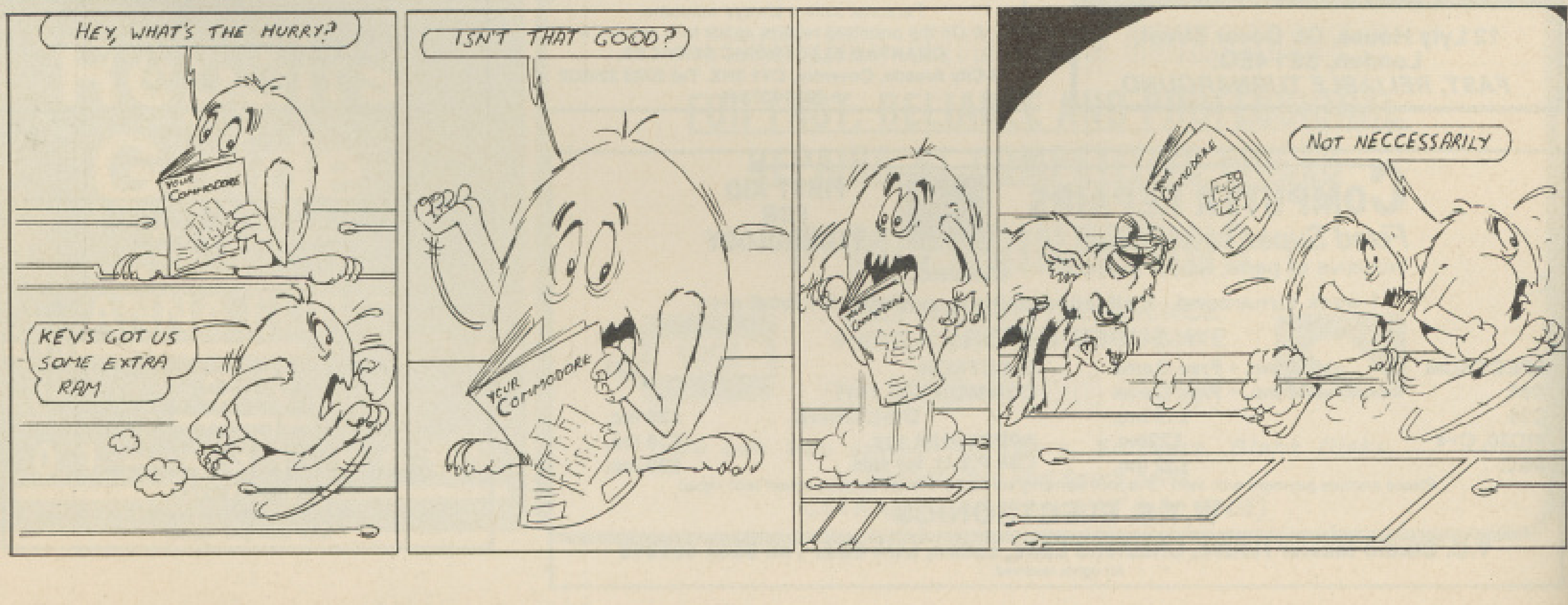
In order to get the program to run correctly you should follow the procedure below:

1) Make sure that you have the 6 Basic loaders to hand.
2) If using cassette make sure that you have a new one handy and that it is a long one.
3) Type in and SAVE the program below onto your disk or onto the cassette mentioned in 2 above. If using cassette change the ,8's to ,1's.
4) Turn OFF and ON your machine.
5) LOAD the first Basic loader and RUN it. A new program will be saved out once this is finished. If using cassette make sure that you place the new cassette into the recorder before you RUN the loader. The new program should be saved directly after the program you have just entered in 3.
6) Turn OFF and ON your machine.
7) LOAD the next Basic loader and repeat the steps 5 to 7 for all remaining Baic loaders.
8) To start the William Tell program simply LOAD and RUN the first program that you SAVEd. This will then LOAD and RUN the rest of the game.

PROGRAM: WILLIAM TELL

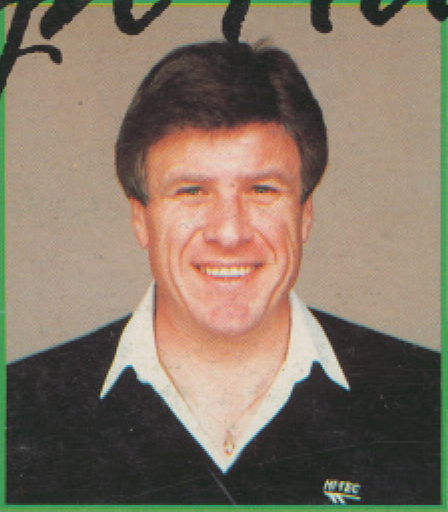
```
10 IFA=0THENLOAD"WT1",8,1:A=A+1
20 IFA=1THENLOAD"WT2",8,1:A=A+1
30 IFA=2THENLOAD"WT3",8,1:A=A+1
40 IFA=3THENLOAD"WT4",8,1:A=A+1
50 IFA=4THENLOAD"WT5",8,1:A=A+1
60 IFA=5THENLOAD"WT6",8,1:A=A+1
70 SYS 35191
```

The Nibbles *By Alan Batchelor*




AMIGA
MICRO Music
THE SPECIALIST
MUSIC MAGAZINE
FOR COMPUTER USERS
ATARI ST
IBM PC
APPLE MAC
KCS-Just what the Doctor ordered!
ACORN
MUSIC X-The Review
MICRO MUSIC
Find out about the magazine the computer music world has been waiting for
COMMODORE 64
AMSTRAD CPC
ZX SPECTRUM

Emlyn Hughes



INTERNATIONAL



Quite simply the best football simulation available for the 64 Nothing short of superb.
ZZAP 64

AVAILABLE NOW FOR COMMODORE 64
£9.95 tape £12.95 disk
Coming soon for Spectrum & Amstrad

Audiogenic

Audiogenic Software Limited, Winchester House, Canning Road, Harrow HA3 7SJ, England

Order by phone on 01 861 1166



Your Commodore March 1989 Volume 5 Number 6

